Das SOPHIST-REgelwerk

Anwendung und Bewertung einer Methode des Requirements Engineerings



Arbeitsberichte zur Wirtschaftsinformatik

Nummer: 02/2008

Autoren: Georg Geltinger, Oliver Kappes, Laura Kolenc, Frank Lehmann

Berufsakademie Ravensburg

Studiengang Wirtschaftsinformatik

Bibliographische Information Der Deutschen Bibliothek

Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über http://dnb.ddb.de abrufbar.

Herausgeber:	Studiengang Wirtschaftsinformatik, Berufsakademie Ravensburg
Anschrift:	Marienplatz 2
	88212 Ravensburg
	+49 (0) 751 / 18999-2731
	e-mail: baechle@ba-ravensburg.de
	http://wiservices.ba-ravensburg.de
Version:	1.0
ISSN:	1865-9616

Inhalt

Abbildungsverzeichnis

Tabellenverzeichnis

1 Überblick und Inhalt

1.1 Vorstellung des Projekts „SOPHIST-REgelwerk“

In der vorliegenden Arbeit werden die Ergebnisse des Projekts „SOPHIST-REgelwerk“ vorgestellt, welches an der Berufsakademie Ravensburg im Wintersemester 2006/07 und Sommersemester 2007 durchgeführt wurde. Beim SOPHIST-REgelwerk handelt es sich um eine Methode des Requirements Engineering, mit der die Qualität von natürlichsprachlichen Anforderungen durch deren gezielte Untersuchung und Korrektur gewährleistet werden soll. Das Regelwerk verfolgt dabei das Ziel, eindeutige, vollständige und widerspruchsfreie Anforderungen zu erzeugen.[1]

Aufgabe des Projekts war es, das Vorgehen des Regelwerks und die Ergebnisse, die mit ihm erzielt werden können, zu bewerten. Ferner sollten die Praxisrelevanz und die Einsetzbarkeit in der Lehre bestimmt werden. Dazu wurde die Anwendung des Regelwerks mithilfe einer simulierten Anforderungserhebung und -analyse für ein fiktives Reisebuchungssystem erprobt.
Die Anwendung und Bewertung des Regelwerks führten drei Studenten des Studiengangs Wirtschaftsinformatik durch. Für die Anforderungserhebung stellte sich ein weiterer Student zur Verfügung, dem die Grundlagen des Regelwerks unbekannt waren.

Ihm sei an dieser Stelle nochmals für seine Teilnahme und seinen Einsatz gedankt, durch den die Durchführung des Projekts erst ermöglicht wurde.

1.2 Gliederung und Aufbau dieser Arbeit

Die Arbeit beginnt in Kap. 2 mit einer kurzen Einführung in das SOPHIST-REgelwerk, in der die Grundlagen vorgestellt werden. Nachfolgend wird in Kap. 3 der Simulationsverlauf der Anforderungsanalyse erläutert, wobei die aufgenommenen Anforderungen, die Anwendung des Regelwerks und die Ergebnisse der Anwendung gezeigt werden. Hierbei erfolgt zudem eine erste Bewertung der Anwendbarkeit und des Nutzens der einzelnen Regeln.

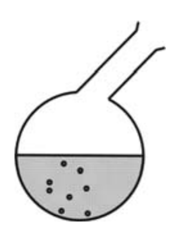
[1] Vgl. (SOPHIST01), S. 3

Darauf aufbauend beschäftigen sich die nachfolgenden Kap. 4 bis 7 mit der Bewertung des Regelwerks nach vier Gesichtspunkten. Begonnen wird in Kap. 4 mit der Untersuchung des Vorgehens. Hierbei wird u. a. geklärt, welchen Aufwand das Regelwerk im Erlernen und Anwenden verursacht und inwieweit bei der Anwendung ein methodisches Vorgehen zu erkennen ist. In Kap. 5 folgt die Bewertung der Ergebnisse der Anwendung bzw. der Anforderungen. Diese werden nicht nur eigenständig auf Qualitätskriterien wie Vollständigkeit oder Eindeutigkeit hin untersucht, sondern auch den ursprünglichen Anforderungen gegenübergestellt. In Kap. 6 wird schließlich die Frage beantwortet, inwieweit das Regelwerk auch in der Unternehmenspraxis eingesetzt werden kann. Die Bewertung schließt mit Kap. 7 ab, in dem geprüft wird, ob bzw. wie das Regelwerk in die Lehre an einer Berufsakademie aufgenommen werden kann.

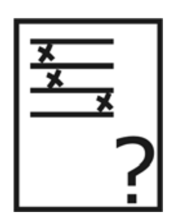
Das der Bewertung nachfolgende Kap. 8 stellt einen Leitfaden für die Anwendung des Regelwerks dar. Dort sind die Erfahrungen, die während der Simulation der Anforderungsanalyse gesammelt wurden, in Form einer Vorlage zusammengefasst.

Das Ende der Arbeit markiert Kap. 9, in dem in einer Zusammenfassung die wichtigsten Vor- und Nachteile des Regelwerks gegenübergestellt werden.

1.3 Benennungs- und Kennzeichnungskonventionen



Die Bewertung des Regelwerks erfolgt anhand der Simulation einer Anforderungserhebung bzw. -analyse. Obgleich viele der Erkenntnisse aus der Simulation unverändert oder lediglich leicht verändert auf eine Anwendung in der Praxis übertragbar sind, existieren auch Stellen, an denen eine Beeinflussung durch die Simulationsumgebung wahrscheinlich ist. Auf diese wird mit dem nebenstehenden Symbol eines Reagenzglases hingewiesen. Zudem wird dort auf Art und Umfang der Beeinflussung eingegangen.



Einige Ergebnisse der Bewertung wurden aus der Befragung des Anforderungsstellers abgeleitet. Diese sind durch das nebenstehende Symbol eines Fragebogens gekennzeichnet.

Die Simulation wurde von drei Studenten durchgeführt, die als Systemanalytiker bzw. Requirements Engineers auftraten und gleichzeitig das Regelwerk bewerteten. Für Sie wird vereinfacht der Begriff „Projektteam" verwendet. Der an der Simulation teilnehmende Student wird gemäß seiner Rolle als „Anforderungssteller" bezeichnet.

2 Einführung in das SOPHIST-REgelwerk

2.1 Grundlagen des SOPHIST-REgelwerks

Das SOPHIST-Regelwerk baut auf Bandlers und Grinders Arbeiten zum Neuro-Linguistischen-Programmieren auf, siehe z. B. (Bandler/Grinder75). Dabei handelt es sich um eine Methode der Psychotherapie, die versucht durch eine gezielte Fragetechnik die Gedankenwelt von Personen aufzudecken. [2]

Bandler und Grinder gehen davon aus, dass Menschen ihre Umwelt selektiv wahrnehmen und daraus ein vereinfachtes Modell in ihrer gedanklichen Vorstellung bilden. Dieses wird „persönliche Wirklichkeit" genannt. Der Prozess der Modellbildung heißt Transformation. Äußern sich Menschen über ihre persönliche Wirklichkeit findet eine erneute Transformation bzw. Modellbildung statt, die sich in „sprachlichen Effekten" zeigt.

Die möglichen Vorgänge, die in beiden Transformationen auftreten können, werden von Bandler und Grinder in die drei Kategorien „Tilgung", „Generalisierung" und „Verzerrung" eingeteilt:

- Bei der Tilgung werden nur bestimmte Sachverhalte der Umwelt wahrgenommen bzw. nur bestimmte Teile der persönlichen Wirklichkeit sprachlich geäußert.
- Im Fall der Generalisierung werden Erkenntnisse über einen Teil der persönlichen Wirklichkeit für einen größeren Teil als exemplarisch betrachtet und auf diesen übertragen.
- Bei der Verzerrung findet eine Verfälschung der Wahrnehmung statt.

Jeder der drei Fälle hat zur Folge, dass die persönliche Wirklichkeit bzw. die sprachliche Äußerung nicht mit der Realität übereinstimmen. Die Fragetechnik des Neuro-Linguistischen-Programmierens will es ermöglichen, die Transformation der sprachlichen Äußerung rückgängig zu machen und damit die persönliche Wirklichkeit zu ermitteln.

Das SOPHIST-REgelwerk wendet diese Erkenntnisse des Neuro-Linguistischen-Programmierens auf Anforderungen bzw. Anforderungssteller an, um damit deren „wirkliche" Anforderungen zu bestimmen.

---

[2] Vgl. im Folgenden (Rupp07), S. 140 ff.

2.2 Überblick über das Regelwerk

Das SOPHIST-REgelwerk umfasst 25 Regeln, mit denen Anforderungen untersucht werden können. Die Regeln sind nach den drei Transformationskategorien gegliedert, wobei sie um zusätzliche linguistische Regeln ergänzt wurden, die die Verständlichkeit und Eindeutigkeit von Anforderungen sicherstellen sollen.
Jede Regel besteht aus einem kurzen, einleitenden Satz (im Folgenden „Regelsatz") und einer ausführlicheren Anweisung oder Erläuterung (im Folgenden „Regelanweisung"), die vorgibt, wie der Regelsatz erfüllt werden kann oder wie er begründet ist. Zu einigen Regeln existieren ferner Signalwörter, deren Vorhandensein in Anforderungen auf die Anwendbarkeit der jeweiligen Regeln hindeutet. Eine Auflistung der Regeln ist im Anhang A zu finden. Für eine ausführliche Darstellung mit Beispielen und zusätzlichen Erläuterungen sei auf (Rupp07) verwiesen.

Um die im Anschluss gezeigte Anwendung im Analyseprozess nachvollziehen zu können, sei darauf hingewiesen, dass das Regelwerk in der Simulation teilweise etwas freier angewendet werden musste. So wurde bspw. das Vorhandensein von Signalwörtern in Anforderungen als optional erachtet. Auf die Interpretation der Regeln durch das Projektteam wird ausführlich in Kap. 3.7 eingegangen. Sollten nachfolgend die Anwendung einer Regel oder die Ergebnisse daraus unklar erscheinen, kann dort nachgeschlagen werden.

In (Rupp07) wird darauf hingewiesen, dass Anforderungen nur soweit verbessert werden sollen bis ihre Qualität für die Weiterverarbeitung ausreicht. Hierbei soll die Anwendung einzelner Regeln von einer Nutzenanalyse abhängig gemacht werden.[3] In der Simulation konnte diese Empfehlung nicht umgesetzt werden, da eine Abschätzung des Risikos, das aus der Nichtanwendung einer Regel entsteht nicht möglich war. Hierbei bleibt anzumerken, dass es auch in der Praxis schwer fallen dürfte, das Schadenspotential aus der Nicht-Anwendung einer Regel zu bestimmen. Das Regelwerk wurde daher vollständig angewendet.

---

[3] Vgl. (Rupp07), S. 147

3 Anwendung des SOPHIST-REgelwerks

3.1 Anforderungserhebung und Simulationsverlauf

Die Anforderungserhebung wurde zunächst mit zwei Studenten durchgeführt. Im ersten Fall handelte es sich um einen Studenten des Studiengangs Hotel-Management, der ein Hotel-Informationssystem spezifizierte. Im zweiten Fall nahm ein Student der Wirtschaftsinformatik mit der Vertiefung Tourismus teil, der die Anforderungen für ein Reisebuchungssystem formulierte. Beide Studenten wurden nicht über den Inhalt des Regelwerks informiert, um einer Beeinflussung vorzubeugen.

In beiden Anforderungserhebungen wurde zunächst das Fachgebiet bzw. die Praxiserfahrung des betreffenden Studenten bestimmt und daraus ein Teilbereich eines möglichen, zu spezifizierenden Systems abgeleitet. Sobald diese Eingrenzung vorgenommen war, stellte das Projektteam allgemeine Fragen zur gewünschten Funktionalität und aufbauend auf den Antworten genauere Fragen. Die Interviews wurden jeweils auf Band aufgenommen und sollten im Nachhinein schriftlich fixiert werden.

In beiden Fällen zeigte sich, dass von den Anforderungsstellern zunächst bereits existierende Systeme oder Anwendungsfälle *beschrieben* wurden anstatt dass Anforderungen an ein neues System gestellt wurden. Ferner war zu erkennen, dass sehr große logische Sprünge in den Schilderungen vorhanden waren. Trotz Beispielen, die zeigen sollten, wie Anforderungen aufgebaut sind und Interventionen bei Abschweifungen, gelang es nicht, die Funktionalität eines neuen Systems durchgängig zu spezifizieren.

Der Anforderungssteller des Hotel-Informationssystems wurde daher gebeten, seine Anforderungen schriftlich zu verfassen, wobei ihm als Leitfaden einige Fragen zur Beantwortung vorgegeben wurden.

Im Fall des Reisebuchungssystems wurde hingegen zusammen mit dem Anforderungssteller ein neues System grob entworfen und die Anforderungen durch einen grafisch dargestellten Arbeitsablauf vorstrukturiert. Mit dieser

Hilfestellung gelang es strukturiertere Anforderungen aufzunehmen und im Anschluss schriftlich niederzulegen.

Nachdem das Anforderungsdokument für das Hotel-Informationssystem den Anforderungen des Reisebuchungssystems in etwa glich, wurde aus organisatorischen Gründen die Fortführung der Anforderungsanalyse für das Reisebuchungssystem beschlossen.

3.2 Erhobene Originalanforderungen

Die ursprünglichen Anforderungen für das Reisebuchungssystem sind im Nachfolgenden dargestellt. Bei dem spezifizierten System handelt es sich um eine Client-Anwendung, die auf einen Server zugreift. Auf diesem Server hinterlegen Reiseveranstalter ihre Reiseangebote. Außerdem stellt der Server die Möglichkeit bereit, die Reisen zu durchsuchen und zu buchen.

1. Um eine Reise zu buchen, muss zunächst erfasst werden, was für eine Reise der Kunde haben will.
2. Zuerst wird der Reisezeitraum festgelegt, danach wird die Anzahl der Personen festgelegt und ob Kinder teilnehmen.
3. Sind Kinder dabei, so muss deren Alter angeben werden.
4. Als Zusatzoptionen kann noch dazu der Abflugs- und Ankunftsflughafen, die Reiseart, das Reiseziel und der Reiseort angegeben werden.
5. Es können auch noch Präferenzen des Kunden angegeben werden.
6. Eine Reiseart kann sein, ob der Kunde einen Badeurlaub oder eine Studienfahrt will.
7. Zusätzlich kann noch eine Hotelkategorie, der Preis und die Verpflegungsart ausgesucht werden.
8. Nach der Selektierung nach den Daten des Kunden gibt das System eine Ergebnisliste aus mit den vorhandenen Hotels mit genau den Kriterien.
9. Von dieser Liste wählt sich der Kunde ein Hotel aus.
10. Es gibt eine Detailanzeige, die dem Kunden dann genauere Informationen zu dem Hotel geben kann.
11. Genauere Informationen können zum Beispiel die Entfernung zum Strand oder die Größe des Hotels sein.

12. Nachdem der Kunde weiß welches Hotel und welche Reise er will, kann die Reise gebucht werden.

13. Bei der Reisebuchung selbst können noch Zusatzoptionen hinzugefügt werden.

14. Diese Zusatzoptionen können eine Reiserücktrittsversicherung, eine Auslandsversicherung, oder Angebote des Hotels, wie zum Beispiel Massagen sein.

15. Es können auch weitere unverbindliche Wünsche des Kunden angegeben werden.

16. Danach müssen die Kundendaten erfasst werden.

17. Die Daten des Reiseanmelders - wie die Adresse oder das Alter - können aus einer Datenbank herausgeholt werden.

18. Der Reiseanmelder ist auch derjenige, der später die Rechnung bezahlt.

19. Es müssen dann noch die Reisenden angegeben werden.

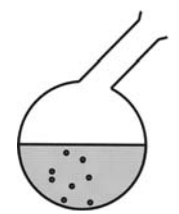
20. Dabei muss dann der Name, das Geschlecht und bei Kindern das Alter angegeben werden.

21. Die Kontodaten des Kunden dürfen aus rechtlichen Gründen nicht mitgespeichert werden.

22. Der Kunde kann dann noch eine Zahlungsart auswählen.

23. Als Zahlungsarten gibt es Zahlung per Lastschrift, mit Kreditkarte oder per Rechnung.

24. Danach kann die Buchung abgeschickt werden und geht dann zum Reiseveranstalter über.



Wie zu erkennen ist, herrscht weiter das *Beschreiben* statt das *Anfordern* im Text vor („Zuerst wird der Reisezeitraum festgelegt, danach wird die Anzahl der Personen festgelegt und ob Kinder teilnehmen.“). Inwieweit dies auf die Simulationsumgebung zurückzuführen ist, kann nicht bestimmt werden. Dass es sich um ein allgemeines Phänomen handelt, lässt sich daraus herleiten, dass beide Anforderungssteller trotz mehrfachen Hinweisen des Projektteams weiter in dieser Ausdrucksform verblieben. Ebenso werden in (Rupp07) Anforderungen in ähnlicher Form als Beispiele für die Anwendung des Regelwerks gegeben. Zudem wird in (Rupp07) die Szenario-Beschreibung („Herr Müller

gibt Daten ein …"), die vergleichbar mit den vorliegenden Anforderungen ist, als besonders leicht verständliche Dokumentations- und Erhebungstechnik vorgestellt. Das Projektteam geht daher davon aus, dass es Anforderungsstellern auch unter realen Bedingungen leichter fällt, Anforderungen „beschreibend" zu äußern.

Neben der beschreibenden Form, dürfte sich ferner der personentypische Sprachgebrauch (bspw. „ausschweifend" vs. „knapp") eines Anforderungsstellers auf die Anwendung des Regelwerks auswirken. Ebenso ist ein Unterschied zwischen schriftlichen und mündlichen Anforderungen zu erwarten, wie auch zwischen kurzen, einfachen und langen, komplexen Anforderungsdokumenten. Die nachfolgende Darstellung versucht auf diese Einflüsse hinzuweisen. Es sei jedoch darauf aufmerksam gemacht, dass es sich dabei nur um Annahmen handeln kann, deren Bestätigung durch weitere Versuche im Rahmen des Projekts nicht erbracht werden kann.

3.3 Vorgehensmodell für die Anwendung

Da in (Rupp07) keine Vorgaben oder Empfehlungen über das Vorgehen gemacht werden, mit dem das Regelwerk in einem Analyseprozess anzuwenden ist, wurde durch das Projektteam ein eigenes Vorgehensmodell entwickelt.

1. Zunächst werden die Anforderungen mit dem Anforderungssteller aufgenommen. Im Ergebnis soll ein schriftliches Anforderungsdokument stehen. Die Erhebung kann mündlich erfolgen und im Anschluss schriftlich fixiert werden oder schriftlich durchgeführt werden.
2. In Abwesenheit des Anforderungsstellers werden die Anforderungen mit dem Regelwerk analysiert. Dabei werden alle Umformungen, die mit plausiblen Annahmen getätigt werden können, sofort vorgenommen. Eine direkte Rücksprache beim Anforderungssteller für jede Umformung wäre nicht praktikabel (sowohl in der Simulation als auch im Praxiseinsatz). Dort wo keine plausiblen Annahmen getroffen werden können, werden die sich ergebenden Fragen erfasst.
3. Der nächste Schritt erfolgt unter Einbeziehung des Anforderungsstellers. Ihm werden die ursprünglichen, sowie die umgeformten Anfor-

derungen zur Kontrolle vorgelegt. Dabei findet eine Bestätigung bzw. Zurückweisung der Annahmen des Teams statt.

Ebenfalls werden die sich ergebenden Fragen geklärt, wobei die Antworten lediglich grob dokumentiert werden und keine sofortige Umformulierung der Anforderungen stattfindet. Ergeben sich beim Klären der Fragen neue Anforderungen, versorgt sich das Projektteam mit ausreichenden Informationen, um diese in der Nachbearbeitung selbst formulieren zu können.

Eine sofortige Neuformulierung zusammen mit dem Anforderungssteller wäre zu zeitaufwendig und würde die Akzeptanz des Regelwerks in starkem Masse gefährden.

4. Sofern sich in Schritt 3 kein Änderungsbedarf für die Anforderungen ergibt, gelten diese als bestätigt und der Analyseprozess endet.
5. Ansonsten werden in der Nachbearbeitung in Abwesenheit des Anforderungsstellers die Antworten in die Anforderungen eingearbeitet und neue Anforderungen formuliert. Bei der Formulierung wird dabei durch das Projektteam darauf geachtet, dass das Regelwerk eingehalten wird. Dies geschieht durch eine Analyse und Korrektur der selbst formulierten Anforderungen mit dem Regelwerk, wobei analog zu Schritt 2 für fehlende Informationen plausible Annahmen getroffen werden und dort wo dies nicht möglich ist, Fragen notiert werden.
6. Die Iteration ist damit abgeschlossen. Der Analyseprozess wird mit den neuen und veränderten Anforderungen in Schritt 3 fortgeführt.

Anmerkung: Im weiteren Verlauf der Arbeit wird eine Anforderung der Iteration zugeordnet, in der die letzten Änderungen vorgenommen wurden.

3.4 Dokumentationssystematik der Anwendung

Die Anwendung des Regelwerks auf die Anforderungen nach dem soeben beschriebenen Vorgehensmodell wird mit folgendem Schema dokumentiert.

Nr.		Iteration
O	**Originalanforderung**	
U	Anwendbare Regeln für Umformungen	
E	**Ergebnis der Umformung**	
F	Anwendbare Regeln für Fragen	
E	**Ergebnis der Rückfragen**	

Den Feldern kommt dabei folgende Bedeutung zu:

Nr.	Jede Anforderung erhält eine eindeutige fortlaufende Nummer.
Iteration	Zeigt die Iteration gemäß der getroffenen Festlegung auf.
O	Originalanforderung des Anforderungsstellers.
U	Regeln, mit denen die Anforderungen unter plausiblen Annahmen umgeformt wurden.
F	Regeln, die Rückfragen beim Anforderungssteller verursachten.
E	Ergebnis aus Umformungen und Einarbeiten von Antworten.

Die Angabe der Regeln erfolgt gemäß der Nummerierung in (Rupp07) bzw. Anhang A. Die Nummer der Regel wird durch einen Buchstaben ergänzt, der die Kategorie der Regel angibt (T: Tilgung, G: Generalisierung, V: Verzerrung, W: Weitere).

3.5 Anwendung des Regelwerks

Nachfolgend werden nun die Iterationen der Anwendung dargestellt. Um den Überblick zu wahren und den Verlauf des Analyseprozesses besser zu verdeutlichen, werden neue Anforderungen, die inhaltlich eng mit einer alten Anforderung verbunden sind, in der Tabelle der alten Anforderung dargestellt.

Ist kein enger Bezug zu alten Anforderungen gegeben oder beziehen sich neue Anforderungen auf mehrere alte Anforderungen, wird eine neue Tabelle am Ende der Liste angelegt.
Eine thematisch und zeitlich geordnete Darstellung der sich ergebenden Anforderungen folgt im nachfolgenden Kap. 3.6. Dort befinden sich zudem die Glossareinträge, die im Analyseprozess entstanden sind.

Nr. 1	Iteration 1
O	**Um eine Reise zu buchen, muss zunächst erfasst werden, was für eine Reise der Kunde haben will.**
U	1T: Formulieren Sie jede Anforderung im Aktiv Die Anforderung befindet sich im Passiv. 21W: Überprüfen Sie Nebensätze „Um eine Reise zu buchen" ist nur eine Begründung.
E	**Das System muss es zunächst ermöglichen zu erfassen, was für eine Reise der Kunde haben will.** **Kommentar: Um eine Reise zu buchen.**
F	3T: Decken Sie unvollständig spezifizierte Prozesswörter auf 5T: Klären Sie Mögliches und Unmögliches Wer erfasst? Wie wird erfasst? → Der Sachbearbeiter im Reisebüro. → Über eine Bildschirmmaske; zum Ankreuzen. 7T: Finden Sie implizite Annahmen Gibt es eine Liste mit Reisen? → Ja, auf einem Server im Internet. Siehe dazu die Ergänzung der Anforderung 8. 12G: Hinterfragen Sie Substantive ohne Bezugsindex Welchem Kunden? → Dem Kunden im Reisebüro, der bei einem Sachbearbeiter eine Reise buchen möchte. 9G: Bestimmen Sie die Universalquantoren Für wen gilt dies? → Jeder Sachbearbeiter (auch in den folgenden Anforderungen). 23W: Definieren Sie Substantive nach folgendem Schema Definiere Reise! Definiere Kunde! → Es wurde geklärt, dass der Begriff „Reise" für verschiedene Leistungen steht (siehe Glossar) und mit der Anforderung eigentlich. eine Auswahlmöglichkeit für die Suche nach diesen Leistungen gewünscht war. → Definition „Kunde" siehe Glossar.
E	**1. Das System besitzt genau eine Einstiegsmaske.** **2. Das System ermöglicht jedem Sachbearbeiter im Reisebüro über je ein Ankreuzfeld auf der Einstiegsmaske zwischen der Suche nach entweder Hotels oder Flügen oder Pauschalreisen zu wählen.** **Kommentar: Je nach Wunsch des Kunden**

Reduzierung des Umfangs
Die weiteren Anforderungen beschränken sich auf die Leistung „Pauschalreise". Anforderungen für das Buchen von Flügen oder einzelnen Hotelaufenthalten werden nicht berücksichtigt.

Nr. 2	Iteration 1
O	**Zuerst wird der Reisezeitraum festgelegt, danach wird die Anzahl der Personen festgelegt und ob Kinder teilnehmen.**
U	1T: Formulieren Sie jede Anforderung im Aktiv Die Anforderung befindet sich im Passiv. 12G: Hinterfragen Sie Substantive ohne Bezugsindex „Personen" sind nicht genauer spezifiziert. Ergänzung: „an der gesuchten Reise teilnehmenden". 22W: Stellen Sie zeitliche Abhängigkeiten klar Es wird ein zeitlicher Ablauf dargestellt.
E	**Wenn der Reisezeitraum der gesuchten Reise festgelegt wurde, dann ermöglicht das System die Anzahl der an der gesuchten Reise teilnehmenden Personen festzulegen und ob Kinder an der gesuchten Reise teilnehmen.**
F	In der ursprünglichen Anforderung wurde ein Zeitverlauf erkannt und nach dem Regelwerk formuliert. Der Anforderungssteller bemerkte dies selbständig und korrigierte, dass eine zeitliche Abfolge in Wirklichkeit nicht gegeben ist. 3T: Decken Sie unvollständig spezifizierte Prozesswörter auf 5T: Klären Sie Mögliches und Unmögliches Wem wird das ermöglicht? Wie wird festgelegt? Was wird festgelegt? → Dem Sachbearbeiter über eine Suchmaske. → Eingabe des Start- und Enddatums, sowie der Zahl der Erwachsenen und Kinder in Eingabefelder. → Als neue Anforderung wurde aufgenommen, dass die Eingabefelder des Reisezeitraums mit Default-Werten belegt sein soll. 14G: Machen Sie konkrete Angaben über die Menge an Objekten Wie werden Reisezeiträume und Personenzahlen festgelegt? Welche Menge an Personen und Kindern darf eingegeben werden? → 1 Reisezeitraum, 1 Anzahl Kinder, 1 Anzahl Personen → 1 <= Personen <= 6 → 0 <= Kinder <=3 23W: Definieren Sie Substantive nach folgendem Schema Definiere Reisezeitraum! → Bei der Definition von „Reisezeitraum" wurde erkannt, dass eigtl. der früheste Starttermin und der späteste Endtermin ausgewählt werden soll. Siehe Glossar. Dadurch wurde erkannt, dass eine Eingabemöglichkeit für die Reisedauer fehlt. Definition „Reisedauer" siehe Glossar.
E	**1. Das System soll jedem Sachbearbeiter genau eine Suchmaske für die Eingabe aller Suchkriterien von Pauschalreisen bereitstellen.** **2. Das System ermöglicht jedem Sachbearbeiter über je ein Datumseingabefeld in der Suchmaske den frühesten Start- und den spätesten Endtermin für die gesuchten Pauschalreisen festzulegen.** **3. Das System belegt das Eingabefeld für den Starttermin mit**

	dem heutigen Datum plus 3 Tage und das Eingabefeld für den Endtermin mit dem heutigen Datum plus 24 Tage. **4. Das System ermöglicht jedem Sachbearbeiter über genau ein verpflichtendes Zahleingabefeld in der Suchmaske die Reisedauer der gesuchten Pauschalreisen in Tagen einzugeben.** **5. Das System belegt das Zahleingabefeld für die Reisedauer mit 7.** **6. Das System ermöglicht jedem Sachbearbeiter über je genau ein Auswahlfeld in der Suchmaske die Anzahl der an den gesuchten Pauschalreisen mitreisenden Erwachsenen und Kinder festzulegen.** **7. Das System belegt das Auswahlfeld für die Anzahl der mitreisenden Erwachsenen mit 1 und das Auswahlfeld für die Anzahl der mitreisenden Kinder mit 0.** **8. Die Anzahl der mitreisenden Erwachsenen muss zwischen 1 und 6 liegen und die Anzahl der Kinder zwischen 0 bis 3.**
Nr. 2	Iteration 2
F	6T: Prüfen Sie Modaloperatoren der Notwendigkeit auf benötigte Ergänzungen Was passiert, wenn keine Werte in den Eingabefeldern eingeben werden? → Bei der Suche sollen die Vorbelegungen verwendet werden. Dies wird in der Anforderung 8 berücksichtigt. Die Validation mit dem Anforderungssteller ergab folgendes: → Für die Eingabe der Reisedauer sind zwei Eingabefelder nötig. Hier sollen die minimale und die maximale Reisedauer eingegeben werden können. Dies wirkt sich auf die Anforderungen 2.4, 2.5, 8.2 und 8.3 aus. → Das Feld für die minimale Reisedauer sollte mit 0 und das Feld für die maximale Reisedauer sollte mit 7 vorbelegt sein. → Des Weiteren ergab sich, dass der Begriff „mitreisend“ ersetzt werden muss durch den Begriff „reisend“, da dieser explizit die Gesamtanzahl der Reisenden umfasst. → Das Feld „mitreisende Erwachsene“ wird geändert in das Auswahllistenfeld „Gesamtzahl Reisende“. Es enthält eine Auswahlliste mit den Werten 1 bis 6. Dadurch wird festgelegt, dass an einer Reise maximal 6 Personen teilnehmen können. Die Gesamtanzahl beinhaltet die teilnehmenden Kinder. → Für die Angabe der Alter der Kinder werden drei weitere Auswahllisten benötigt. Damit entfällt Anforderung 3. Durch Berechnungen kann die Anzahl der Kinder ermittelt werden. → Der Anforderungssteller erkannte, dass die Vorbelegung des Feldes für die Gesamtanzahl der Reisenden 2 sein muss.
E	**1. Das System soll jedem Sachbearbeiter genau eine Suchmaske für die Eingabe aller Suchkriterien von Pauschalreisen bereitstellen.**

	2. Das System ermöglicht jedem Sachbearbeiter über je ein Datumseingabefeld in der Suchmaske den frühesten Start- und den spätesten Endtermin für die gesuchten Pauschalreisen festzulegen. **3. Das System belegt das Eingabefeld für den Starttermin mit dem heutigen Datum plus 3 Tage und das Eingabefeld für den Endtermin mit dem heutigen Datum plus 24 Tage.** **4. Das System ermöglicht jedem Sachbearbeiter über je genau ein Zahleingabefeld in der Suchmaske die minimale Reisedauer und die maximale Reisedauer der gesuchten Pauschalreisen in Tagen einzugeben.** **5. Das System belegt das Zahleingabefeld für die minimale Reisedauer mit 0 und das Zahleingabefeld für die maximale Reisedauer mit 7.** **6. Das System ermöglicht jedem Sachbearbeiter über genau ein Auswahlfeld in der Suchmaske die Gesamtanzahl der an den gesuchten Pauschalreisen reisenden Personen festzulegen.** **Kommentar: Die reisenden Personen umfassen die reisenden Erwachsenen und die reisenden Kinder.** **7. Das System belegt das Auswahlfeld für die Gesamtanzahl der reisenden Personen mit 2.** **8. Das System ermöglicht jedem Sachbearbeiter das Alter jedes reisenden Kindes über genau eine der drei Auswahllisten anzugeben.** **9. Jede Auswahlliste für das Alter der reisenden Kinder enthält die Werte von 0 bis 16 und den Wert „–“.** **10. Das System belegt jede Auswahlliste für das Alter der reisenden Kinder mit dem Wert „-“ vor.** **11. Die Anzahl der reisenden Personen muss zwischen 1 und 6 liegen.** **12. Das System berechnet die Anzahl der reisenden Kinder aus der Anzahl der nicht mit „-“ belegten Auswahllisten für das Alter der reisenden Kinder.** **13. Das System berechnet die Anzahl der reisenden Erwachsenen aus der Gesamtanzahl der reisenden Personen minus der Anzahl der reisenden Kinder.**

Nr. 3		Iteration 1
O	**Sind Kinder dabei, so muss deren Alter angeben werden.**	
U	1T: Formulieren Sie jede Anforderung im Aktiv Die Anforderung befindet sich im Passiv. 12G: Hinterfragen Sie Substantive ohne Bezugsindex „Kinder“ wird nicht genauer spezifiziert. Ergänzung: „an der gewünschten Reise teilnehmende“. 13G: Verwenden Sie Substantive stets in der Einzahl Kinder steht im Plural. 14G: Machen Sie konkrete Angaben über die Menge an Objekten Die Menge der Kinder ist nicht spezifiziert.	
E	**Für jedes an der gewünschten Reise teilnehmende Kind, muss es das System ermöglichen dessen Alter anzugeben.**	
F	3T: Decken Sie unvollständig spezifizierte Prozesswörter auf 5T: Klären Sie Mögliches und Unmögliches Wem wird das ermöglicht? Wie? → Dem Sachbearbeiter im Reisebüro. → Durch verpflichtende Eingabefelder. 23W: Definieren Sie Substantive nach folgendem Schema Definiere Kind! → Siehe Glossar.	
E	**Das System ermöglicht jedem Sachbearbeiter über je ein verpflichtendes Eingabefeld in der Suchmaske das Alter jedes an den gesuchten Pauschalreisen teilnehmenden Kindes anzugeben.**	
Nr. 3		Iteration 2
F	6T: Prüfen Sie Modaloperatoren der Notwendigkeit auf benötigte Ergänzungen Wie soll die Verpflichtung aussehen? → Die Angabe der Anzahl der Kinder wird durch die Angabe des Alters der Kinder geregelt. Siehe Anforderung 2.	
E	**Anforderung 3 entfällt; die Anforderung 3 wird durch die Änderungen in Iteration 2 in Anforderung 2 abgedeckt.**	

Nr. 4	Iteration 1
O	**Als Zusatzoptionen kann noch dazu der Abflugs- und Ankunftsflughafen, die Reiseart, das Reiseziel und der Reiseort angegeben werden.**
U	1T: Formulieren Sie jede Anforderung im Aktiv Die Anforderung befindet sich im Passiv. 12G: Hinterfragen Sie Substantive ohne Bezugsindex Die Aufzählung der Zusatzoptionen ist nicht genauer spezifiziert. Ergänzung: „Der gewünschten Reise". 20W: Vermeiden Sie Floskeln „Noch dazu" ist überflüssig.
E	**Das System ermöglicht es als Zusatzoptionen den Abflugs- und Ankunftsflughafen, die Reiseart, das Reiseziel und den Reiseort der gewünschten Reise anzugeben.**
F	3T: Decken Sie unvollständig spezifizierte Prozesswörter auf, 5T: Klären Sie Mögliches und Unmögliches Wem wird das ermöglicht? Wie wird angegeben? → Dem Sachbearbeiter im Reisebüro. → Über Auswahllisten. 7T: Finden Sie implizite Annahmen Gibt es eine Liste der aufgeführten Zusatzoptionen? → Ja sie wird mit dem CRS-Server abgeglichen. Siehe Anforderung 25 und Glossar. 14G: Machen Sie konkrete Angaben über die Menge an Objekten Welche Menge an Zusatzoptionen kann je angegeben werden? → Für jede Zusatzoption können einzelne Werte, mehrere Werte oder auch alle Werte ausgewählt werden. Es können keine bis alle Optionen angegeben werden. 23W: Definieren Sie Substantive nach folgendem Schema Definiere Reiseart, Reiseziel und Reiseort, Abflugs- und Ankunftsflughafen! → Siehe Glossar. Durch Klärung der Begriffe ergab sich, dass Reiseart und Präferenz synonym verwendet werden (Es wird hierüber jeweils bezeichnet, ob es sich um beispielsweise ein Strandhotel, ein Wellness-Hotel oder ein Sporthotel handelt). Anforderung 5 entfällt somit und Anforderung 6 wird als Glossareintrag abgeändert. Durch die Begriffsklärung wurde erkannt, dass die Auswahl der Reisekategorien fehlt. Unter einer Reisekategorie versteht der Anforderungssteller beispielsweise einen Badeurlaub, eine Wellnessreise oder einen Golfurlaub. Siehe dazu auch Glossar.
E	**Das System ermöglicht jedem Sachbearbeiter je null bis alle Einträge aus je genau einer Auswahlliste für den Abflugs- und Ankunftsflughafen, die Reisekategorie, die Reiseart, das Reiseziel und den Reiseort der zu suchenden Pauschalreise auszuwählen.**

Nr. 5	Iteration 1
O	**Es können auch noch Präferenzen des Kunden angegeben werden.**
U	1T: Formulieren Sie jede Anforderung im Aktiv Die Anforderung befindet sich im Passiv. 12G: Hinterfragen Sie Substantive ohne Bezugsindex „Präferenzen" ist nicht genauer spezifiziert. Ergänzung: „Der gewünschten Reise".
E	**Das System ermöglicht es auch die Präferenzen des Kunden für die gewünschte Reise anzugeben.**
F	3T: Decken Sie unvollständig spezifizierte Prozesswörter auf 5T: Klären Sie Mögliches und Unmögliches Wem wird es ermöglicht? Wie wird angegeben? 14G: Machen Sie konkrete Angaben über die Menge an Objekten Welche Menge an Präferenzen? 23W: Definieren Sie Substantive nach folgendem Schema Definiere Präferenzen! → Der Begriff „Präferenzen" wurde als Synonym für „Reiseart" erkannt.
E	**Anforderung 5 entfällt (siehe Begründung in Anforderung 4).**

Nr. 6	Iteration 1
O	**Eine Reiseart kann sein, ob der Kunde einen Badeurlaub oder eine Studienfahrt will.**
U	
E	
F	Hier handelt sich um keine Anforderung. Es muss geklärt werden, ob eine Anforderung daraus entsteht oder ob es ein Kommentar oder Lexikoneintrag ist.
E	**Begriff wurde mit dem Anforderungssteller genauer bestimmt und in das Glossar übernommen. Anforderung 6 entfällt (siehe Begründung in Anforderung 4).**

Nr. 7	Iteration 1
O	**Zusätzlich kann noch eine Hotelkategorie, der Preis und die Verpflegungsart ausgesucht werden.**
U	1T: Formulieren Sie jede Anforderung im Aktiv Die Anforderung befindet sich im Passiv. 12G: Hinterfragen Sie Substantive ohne Bezugsindex Die Aufzählung ist nicht genauer spezifiziert. Ergänzung: „der gewünschten Reise“.
E	**Das System ermöglicht es eine Hotelkategorie, den Preis und die Verpflegungsart der gewünschten Reise anzugeben.**
F	3T: Decken Sie unvollständig spezifizierte Prozesswörter auf 5T: Klären Sie Mögliches und Unmögliches Wem wird es ermöglicht? Wie wird angegeben? → Dem Sachbearbeiter im Reisebüro. → Hotelkategorie und Verpflegungsart über Auswahllisten. → Preis über Eingabefelder. 14G: Machen Sie konkrete Angaben über die Menge an Objekten Welche Menge an Auswahloptionen? → Bei Hotelkategorie und Verpflegungsart kann je ein oder kein Wert angegeben werden. → Bei Preis kann je Ober- und Untergrenze angegeben werden. 7T: Finden Sie implizite Annahmen Gibt es eine Liste mit Hotelkategorien, Preisen und Verpflegungsarten? → Ja sie wird mit dem CRS Server abgeglichen. Siehe Anforderung 25. 23W: Definieren Sie Substantive nach folgendem Schema Definiere Hotelkategorien, Preise und Verpflegungsart! → Siehe Glossar.
E	**1. Das System ermöglicht jedem Sachbearbeiter über je genau eine Auswahlliste in der Suchmaske optional genau eine Hotelkategorie und optional genau eine Verpflegungsart für alle zu suchenden Pauschalreisen auszuwählen.** **2. Das System ermöglicht jedem Sachbearbeiter über zwei Zahleingabefelder in der Suchmaske optional die Untergrenze und optional die Obergrenze für den Preis aller zu suchender Pauschalreisen anzugeben.**
Nr. 7	Iteration 2
F	Bei der Validation von Anforderung 8 erkannte der Anforderungssteller, dass doch mehrere Verpflegungsarten angegeben werden können.
E	**1. Das System ermöglicht jedem Sachbearbeiter über genau eine Auswahlliste in der Suchmaske optional genau eine Hotelkategorie für alle zu suchenden Pauschalreisen auszuwählen.** **2. Das System ermöglicht jedem Sachbearbeiter über genau eine Auswahlliste keine bis alle Verpflegungsarten für alle zu suchenden Pauschalreisen auszuwählen.** **3. Das System ermöglicht jedem Sachbearbeiter über zwei Zahleingabefelder in der Suchmaske optional die Untergrenze**

	und optional die Obergrenze für den Preis aller zu suchender Pauschalreisen anzugeben.

Reduzierung des Umfangs
Auf die genauere Beschreibung der Kommunikation und der Vernetzung des Systems mit dem CRS-Server wird hier aus Komplexitätsgründen verzichtet.

Nr. 8	Iteration 1
O	**Nach der Selektierung nach den Daten des Kunden gibt das System eine Ergebnisliste aus mit den vorhandenen Hotels mit genau den Kriterien.**
U	2T: Drücken Sie Prozesse durch Vollverben aus, 17V: Hinterfragen Sie Nominalisierungen „Selektierung" ist kein Vollverb, sondern eine Nominalisierung 14G: Machen Sie konkrete Angaben über die Menge an Objekten Die Menge der Hotels wird nicht genauer spezifiziert. Ergänzung: „alle" 22W: Stellen Sie zeitliche Abhängigkeiten klar Es wird ein Ablauf geschildert.
E	**Wenn die Daten des Kunden selektiert wurden, gibt das System genau eine Ergebnisliste aus mit allen vorhandenen Hotels mit genau den Kriterien.**
F	3T: Decken Sie unvollständig spezifizierte Prozesswörter auf 5T: Klären Sie Mögliches und Unmögliches Wem wird die Liste ausgegeben? Wie? Wie sieht sie aus? Woher kommt sie? → Dem Sachbearbeiter im Reisebüro. → Über eine Anzeige auf dem Bildschirm. → Diese Liste ist wie eine Art Explorer aufgebaut. Sie hat rechts eine hierarchische Struktur mit Reiseziel... Im linken Bereich werden dann jeweils die zugehörigen Hotels angezeigt. → Der CRS-Server liefert sie. 7T: Finden Sie implizite Annahmen Gibt es eine Liste mit Hotels und Kriterien? → Ja auf dem CRS-Server. 11G: Ermitteln Sie unvollständig spezifizierte Bedingungen Was passiert, wenn kein Hotel gefunden wird? → Es sollten weniger Kriterien für die Suche angegeben werden. 12G: Hinterfragen Sie Substantive ohne Bezugsindex Welche Daten, Welche Kriterien, Welche Hotels aus welcher Menge? → Daten sind die Suchkriterien für die Reise. → Kriterien sind die in der Suchmaske eingegebenen Suchkriterien.
E	**1. Wenn der Sachbearbeiter die Suchkriterien des Kunden für alle zu suchenden Pauschalreisen in die Suchmaske eingegeben hat und auf den Button „Suche starten" in der Suchmaske gedrückt hat, schickt das System die Suchanfrage an den CRS-Server ab und stellt das vom CRS-Server zurückgelieferte Such-**

	ergebnis in genau einer Ergebnisliste dar. **2. Die Ergebnisliste soll alle auf dem CRS-Server verfügbaren Pauschalreisen** **mit Reisezeitraum innerhalb dem ausgewählten frühesten Starttermin und spätesten Endtermin** **und Preis innerhalb der ausgewählten Untergrenze und Obergrenze** **und ausgewählter Reisedauer** **und ausgewählter Zahl der mitreisenden Erwachsenen und Kindern** **und ausgewählter Hotelkategorie** **und ausgewählter Verpflegungsart** **und einem der ausgewählten Abflugshäfen** **und einem der ausgewählten Ankunftsflughäfen** **und einem der ausgewählten Reiseorte** **und einem der ausgewählten Reiseziele** **und einer der ausgewählten Reisekategorien** **und einer der ausgewählten Reisearten enthalten.** **3. Wenn das Eingabefeld für die Reisedauer keinen Wert enthält, soll die Ergebnisliste alle Pauschalreisen mit der Reisedauer „7 Tage“ enthalten.** **4. Wenn das Eingabefeld für den frühesten Starttermin keinen Wert enthält, soll die Ergebnisliste alle Pauschalreisen mit Starttermin größer oder gleich dem heutigen Datum plus 3 Tage enthalten.** **5. Wenn das Eingabefeld für den spätesten Endtermin keinen Wert enthält, soll die Ergebnisliste alle Pauschalreisen mit Endtermin kleiner oder gleich dem heutigen Datum plus 24 Tage enthalten.** **6. Wenn keine Hotelkategorie** **oder keine Verpflegungsart** **oder keine Untergrenze des Preises** **oder keine Obergrenze des Preises** **oder kein Ankunftsflughafen** **oder kein Abflugsflughafen** **oder kein Reiseort** **oder kein Reiseziel** **oder keine Reisekategorie** **oder keine Reiseart** **ausgewählt wurde, soll jeweils nach allen möglichen Werten jedes nicht gewählten Suchkriteriums gesucht werden.** **7. Wenn die Ergebnisliste keine Ergebniseinträge zu den Suchkriterien enthält, gibt das System dem Sachbearbeiter den Hinweis „Keine Einträge gefunden“ aus, verbleibt in der Suchmaske und ermöglicht dem Sachbearbeiter alle Suchkriterien in der Suchmaske zu ändern.**

<table>
<tr><td></td><td colspan="2">8. Wenn der Sachbearbeiter in der Suchmaske den Button „Suche starten“ gedrückt hat und das System vom CRS nicht innerhalb von 10 Sekunden vom CRS-Server genau eine Ergebnisliste abrufen kann, soll das System dem Sachbearbeiter den Hinweis „CRS-Server nicht erreichbar“ ausgeben.</td></tr>
<tr><td colspan="2">Nr. 8</td><td>Iteration 2</td></tr>
<tr><td>F</td><td colspan="2">In der Validation wurden folgende Punkte erkannt:
→ Aus Anforderung 2 ergibt sich die Umbenennung „mitreisend“ in „reisend“.
→ Es soll auch Reisen mit einer höheren Hotelkategorie gesucht werden können (8.2).
→ Es soll nach mehreren Verpflegungsarten gesucht werden können (8.2).
→ Es muss die minimale und maximale Reisedauer berücksichtigt werden, die mit 0 bzw. 7 vorbelegt sein muss (8.3).</td></tr>
<tr><td>E</td><td colspan="2">1. Wenn der Sachbearbeiter die Suchkriterien des Kunden für alle zu suchenden Pauschalreisen in die Suchmaske eingegeben hat und auf den Button „Suche starten“ in der Suchmaske gedrückt hat, schickt das System die Suchanfrage an den CRS-Server ab und stellt das vom CRS-Server zurückgelieferte Suchergebnis in genau einer Ergebnisliste dar.

2. Die Ergebnisliste soll alle auf dem CRS-Server verfügbaren Pauschalreisen
mit Reisezeitraum innerhalb dem ausgewählten frühesten Starttermin und spätesten Endtermin
und Preis innerhalb der ausgewählten Untergrenze und Obergrenze
und Reisedauer innerhalb der ausgewählten Unter- und Obergrenze
und ausgewählter Zahl der reisenden Erwachsenen und Kindern
und ausgewählter oder höherer als ausgewählter Hotelkategorie
und einer der ausgewählten Verpflegungsarten
und einem der ausgewählten Abflugshäfen
und einem der ausgewählten Ankunftsflughäfen
und einem der ausgewählten Reiseorte
und einem der ausgewählten Reiseziele
und einer der ausgewählten Reisekategorien
und einer der ausgewählten Reisearten enthalten.

3. Wenn das Eingabefeld für die minimale Reisedauer keinen Wert enthält, soll die Ergebnisliste alle Pauschalreisen mit der Reisedauer gleich oder größer „0 Tage“ enthalten.

4. Wenn das Eingabefeld für die maximale Reisedauer keinen Wert enthält, soll die Ergebnisliste alle Pauschalreisen mit der Reisedauer kleiner oder gleich „7 Tage“ enthalten.

5. Wenn das Eingabefeld für den frühesten Starttermin keinen Wert enthält, soll die Ergebnisliste alle Pauschalreisen mit Starttermin größer oder gleich dem heutigen Datum plus 3 Tage enthalten.</td></tr>
</table>

6. Wenn das Eingabefeld für den spätesten Endtermin keinen Wert enthält, soll die Ergebnisliste alle Pauschalreisen mit Endtermin kleiner oder gleich dem heutigen Datum plus 24 Tage enthalten.

7. Wenn keine Hotelkategorie
oder keine Verpflegungsart
oder keine Untergrenze des Preises
oder keine Obergrenze des Preises
oder kein Ankunftsflughafen
oder kein Abflugsflughafen
oder kein Reiseort
oder kein Reiseziel
oder keine Reisekategorie
oder keine Reiseart
ausgewählt wurde, soll jeweils nach allen möglichen Werten jedes nicht gewählten Suchkriteriums gesucht werden.

8. Wenn die Ergebnisliste keine Ergebniseinträge zu den Suchkriterien enthält, gibt das System dem Sachbearbeiter den Hinweis „Keine Einträge gefunden“ aus, verbleibt in der Suchmaske und ermöglicht dem Sachbearbeiter alle Suchkriterien in der Suchmaske zu ändern.

9. Wenn der Sachbearbeiter in der Suchmaske den Button „Suche starten“ gedrückt hat und das System vom CRS nicht innerhalb von 10 Sekunden vom CRS-Server genau eine Ergebnisliste abrufen kann, soll das System dem Sachbearbeiter den Hinweis „CRS-Server nicht erreichbar“ ausgeben und ermöglichen die Suche abzubrechen oder neu zu starten.

Reduzierung des Umfangs
Auf eine detaillierte Beschreibung der Ergebnisliste und der Vernetzung mit dem CRS-Server wird aufgrund der hohen Komplexität verzichtet.

Nr. 9	Iteration 1
O	**Von dieser Liste wählt sich der Kunde ein Hotel aus.**
U	12G: Hinterfragen Sie Substantive ohne Bezugsindex „Liste“ ist nicht genauer spezifiziert. Ergänzung: „Ergebnisliste“. 14G: Machen Sie konkrete Angaben über die Menge an Objekten Es werden keine definierten quantitativen Angaben zu Hotel gemacht. Ergänzung: „genau eines“.
E	**Das System ermöglicht es dem Kunden von der Ergebnisliste genau ein Hotel auszuwählen.**
F	5T: Klären Sie Mögliches und Unmögliches Wie wird das ermöglicht? Wie wird ausgewählt? Für was wird ausgewählt. → Es wurde erkannt, dass in der umformulierten Anforderung der Zusatz „für die Detailanzeige“ fehlt. Der Anforderungssteller erkannte dies selbständig und korrigierte dies. → Über eine Markierung des Hotels. → Nicht der Kunde sondern der Sachbearbeiter erledigt dies. 11G: Ermitteln Sie unvollständig spezifizierte Bedingungen Was passiert, wenn der Kunde keines auswählt? → Das ganze Verfahren wird abgebrochen, oder es muss eine erneute Suche durchgeführt werden. 14G: Machen Sie konkrete Angaben über die Menge an Objekten Welche Hotels können ausgewählt werden? → Jedes aus der Liste.
E	**1. Das System ermöglicht es jedem Sachbearbeiter jedes vom Kunden gewünschte Hotel aus der Ergebnisliste über einen Doppelklick mit der linken Maustaste für genau eine Detailanzeige auszuwählen.** **2. Das System ermöglicht jedem Sachbearbeiter über den Button „Zurück zur Suche“ in der Ergebnisliste zur Suchmaske zurückzukehren.**

Nr. 10		Iteration 1
O	**Es gibt eine Detailanzeige, die dem Kunden dann genauere Informationen zu dem Hotel geben kann.**	
U	14G: Machen Sie konkrete Angaben über die Menge an Objekten Es werden ungenaue Angaben über Kunden und Hotels gemacht. Ergänzung: „genau eine“ bzw. „zu jedem“. 21W: Überprüfen Sie Nebensätze „… die dem Kunden dann genauere Informationen zu dem Hotel geben kann“ stellt eine Begründung dar.	
E	**Das System stellt jedem Kunden genau eine Detailanzeige zu jedem Hotel bereit.** **Kommentar: Sie gibt dem Kunden genauere Informationen zu dem Hotel.**	
F	4T: Ermitteln Sie unvollständige Vergleiche und Steigerungen „genauer“ als was? → Als die Ergebnisliste. Wird zukünftig mit „Detailinformationen“ benannt. Siehe Glossar. 5T: Klären Sie Mögliches und Unmögliches Woher kommen die Informationen? Wie stellt es die Anzeige bereit? → Sie werden vom CRS Server abgeholt, Siehe Anforderung 26. → Über eine neue Anzeigemaske. → Informationen als Fließtext. Siehe Anforderung 26. 7T: Finden Sie implizite Annahmen Gibt es eine Liste mit Detailanzeigen? → Die Detailinformationen kommen vom CRS-System. Siehe neue Anforderung 26. 8T: Überprüfen Sie das Verhältnis der Objekte Welche Beziehung existiert zwischen Hotel und Detailanzeige? Gibt es immer eine Detailanzeige? → Nur wenn genauere Informationen auf dem CRS-Server vorhanden sind. Siehe neue Anforderung 26. 12G: Hinterfragen Sie Substantive ohne Bezugsindex Was sind genauere Informationen? → Entfällt, da dies durch die erste Umformulierung im Kommentar steht.	
E	**Das System stellt jedem Sachbearbeiter genau eine Detailanzeige mit allen Detailinformationen zu jedem Hotel aus der Ergebnisliste für jeden Kunden bereit.**	

Nr. 11	Iteration 1
O	**Genauere Informationen können zum Beispiel die Entfernung zum Strand oder die Größe des Hotels sein.**
U	
E	
F	Hier handelt sich um keine Anforderung. Es muss geklärt werden, ob eine Anforderung daraus entsteht oder ob es ein Kommentar oder Lexikoneintrag ist.
E	**Begriff wurde mit dem Anforderungssteller genauer bestimmt und in das Glossar übernommen. Anforderung 11 entfällt.**

Reduzierung des Umfangs
Auf die genauere Beschreibung der Kommunikation und der Vernetzung des Systems mit dem CRS-Server wird hier aus Komplexitätsgründen verzichtet.

Nr. 12	Iteration 1
O	**Nachdem der Kunde weiß welches Hotel und welche Reise er will, kann die Reise gebucht werden.**
U	1T: Formulieren Sie jede Anforderung im Aktiv Die Anforderung befindet sich im Passiv. 12G: Hinterfragen Sie Substantive ohne Bezugsindex Reise und Hotel sind nicht genauer spezifiziert. Ergänzung: „aus der Ergebnisliste". 14G: Machen Sie konkrete Angaben über die Menge an Objekten Es wird keine konkrete Angabe über die Menge der Reisen gemacht. Ergänzung: „genau eine". 22W: Stellen Sie zeitliche Abhängigkeiten klar Es wird eine Abfolge dargestellt.
E	**Wenn der Kunde weiß welches Hotel und welche Reise aus der Ergebnisliste er will, ermöglicht das System genau eine Reise zu buchen.**
F	3T: Decken Sie unvollständig spezifizierte Prozesswörter auf 5T: Klären Sie Mögliches und Unmögliches Wem wird das? Wie wird gebucht? → Dem Sachbearbeiter im Reisebüro. → Es wird gebucht, indem in der Detailanzeige auf „Buchen" geklickt wird. Zunächst werden weitere Informationen zur Buchung eingegeben. 11G: Ermitteln Sie unvollständig spezifizierte Bedingungen Was ist wenn der Kunde nicht weiß, welche Reise er möchte? → Es muss aus der Gesamtliste zurück gegangen werden können (siehe Anforderung 9). 8T: Überprüfen Sie das Verhältnis der Objekte Wie stehen Reise und Hotel zueinander? → Jeder Pauschalreise ist ein Hotel zugeordnet. Es können zudem mehrere Flüge zu einer Pauschalreise ausgewählt werden. Siehe Anforderung Nr. 27.

	23W: Definieren Sie Substantive nach folgendem Schema Definiere Hotel! → Siehe Glossar. Aus der umformulierten Anforderung kann der Zusatz „welches Hotel und" gestrichen werden. Dies kann über die Definition von „Reise" und „Hotel" erkannt werden. 25W: Definieren Sie Verben nach folgendem Schema Definiere buchen! → Siehe Glossar. „Buchen" wird durch „für Reisebuchung auswählen" ersetzt werden, da das System an dieser Stelle noch nicht die Buchung durchführen sollte, sondern noch Zusatzinformationen nötig sind.
E	**1. Jeder Eintrag in der Ergebnisliste der Pauschalreisen besitzt genau einen Button „zum Buchen".** **2. Das System ermöglicht jedem Sachbearbeiter genau eine Pauschalreise für die Reisebuchung auszuwählen indem er auf den Button „zum Buchen" in dem Eintrag der Ergebnisliste der gewählten Pauschalreisen klickt.** **Kommentar: Sobald der Kunde weiß, welche Pauschalreise aus der Ergebnisliste er will.**

Nr. 13	Iteration 1
O	**Bei der Reisebuchung selbst können noch Zusatzoptionen hinzugefügt werden.**
U	1T: Formulieren Sie jede Anforderung im Aktiv Die Anforderung befindet sich im Passiv. 2T: Drücken Sie Prozesse durch Vollverben aus, 17V: Hinterfragen Sie Nominalisierungen „Reisebuchung" ist kein Vollverb, sondern eine Nominalisierung. 12G: Hinterfragen Sie Substantive ohne Bezugsindex „Zusatzoptionen" ist nicht genauer spezifiziert. Ergänzung: „zur gebuchten Reise". 20W: Vermeiden Sie Floskeln „Noch" ist überflüssig.
E	**Wenn jmd. genau eine Reise bucht, ermöglicht das System Zusatzoptionen zur gebuchten Reise hinzuzufügen.**
F	3T: Decken Sie unvollständig spezifizierte Prozesswörter auf 5T: Klären Sie Mögliches und Unmögliches Wem wird das ermöglicht? Wie wird hinzugefügt? → Dem Sachbearbeiter im Reisebüro. → Über eine Eingabemaske. 7T: Finden Sie implizite Annahmen Gibt es eine Liste mit Zusatzoptionen? → Die Liste mit den Zusatzoptionen wird mit dem CRS-System abgeglichen. Siehe Anforderung 28. 8T: Überprüfen Sie das Verhältnis der Objekte Wie stehen Zusatzoptionen mit einer Reise in Verbindung? → Zusatzoptionen werden zusätzlich zu einer Reise gekauft. Sie sind unterschiedlich je Reise. Siehe Anforderung 28. 14G: Machen Sie konkrete Angaben über die Menge an Objekten Welche Menge an Zusatzoptionen? → Keine Begrenzung. 23W: Definieren Sie Substantive nach folgendem Schema Definiere Zusatzoptionen! Definiere Reisebuchung! → Siehe Glossar.
E	**1. Das System besitzt genau eine Buchungsmaske.** **2. Bei der Reisebuchung ermöglicht das System jedem Sachbearbeiter über je ein Zahleingabefeld in der Buchungsmaske genau eine Bestellmenge zu jeder Zusatzoption des Hotels und zu jeder Zusatzoption des Reiseveranstalters der gewählten Pauschalreise einzugeben.**

Nr. 14		Iteration 1
O	**Diese Zusatzoptionen können eine Reiserücktrittsversicherung, eine Auslandsversicherung, oder Angebote des Hotels, wie zum Beispiel Massagen sein.**	
U		
E		
F	Hier handelt es sich um keine Anforderung. Es muss geklärt werden, ob eine Anforderung daraus entsteht oder ob es ein Kommentar oder Lexikoneintrag ist.	
E	**Anforderung 14 wird als Glossareintrag umgewandelt.**	

Nr. 15		Iteration 1
O	**Es können auch weitere unverbindliche Wünsche des Kunden angegeben werden.**	
U	1T: Formulieren Sie jede Anforderung im Aktiv Die Anforderung befindet sich im Passiv. 20W: Vermeiden Sie Floskeln „Auch“ ist überflüssig.	
E	**Das System ermöglicht weitere unverbindliche Wünsche des Kunden anzugeben.**	
F	3T: Decken Sie unvollständig spezifizierte Prozesswörter auf 5T: Klären Sie Mögliches und Unmögliches Wem wird das ermöglicht? Wie wird angegeben? Wann? → Dem Sachbearbeiter im Reisebüro. → Über ein Freitext-Eingabefeld. → Bei der Buchung der Reise. 14G: Machen Sie konkrete Angaben über die Menge an Objekten Welche Menge an Wünschen? → Begrenzung über die Größe des Freitext-Feldes (50 Zeichen). 23W: Definieren Sie Substantive nach folgendem Schema Definiere Wünsche! → Siehe Glossar. 24W: Definieren Sie Eigenschaftswörter nach folgendem Schema Definiere unverbindlich! → Siehe Glossar.	
E	**Das System ermöglicht jedem Sachbearbeiter optional bei der Reisebuchung weitere unverbindliche Wünsche des Kunden in genau einem 50 Zeichen langem Freitextfeld in der Buchungsmaske anzugeben.**	

Nr. 16	Iteration 1
O	**Danach müssen die Kundendaten erfasst werden.**
U	1T: Formulieren Sie jede Anforderung im Aktiv Die Anforderung befindet sich im Passiv.
E	**Das System muss ermöglichen, die Kundendaten zu erfassen.**
F	3T: Decken Sie unvollständig spezifizierte Prozesswörter auf 5T: Klären Sie Mögliches und Unmögliches Wem wird das ermöglicht? Wie wird erfasst? → Dem Sachbearbeiter im Reisebüro. → Über eine Eingabemaske. 12G: Hinterfragen Sie Substantive ohne Bezugsindex Welche Kundendaten? Von wem? Wie sehen sie aus? → Name, Adresse des Kunden (=Reiseanmelder). 14G: Machen Sie konkrete Angaben über die Menge an Objekten Welche Menge an Kundendaten? → Die Kundendaten genau eines Reiseanmelders. 22W: Stellen Sie zeitliche Abhängigkeiten klar Wonach müssen Kundendaten erfasst werden? → Nach den Zusatzoptionen, Wünschen und Flugverbindungen. 23W: Definieren Sie Substantive nach folgendem Schema Definiere Kundendaten? → Wurde genauer in die Anforderung aufgenommen.
E	**1. Das System besitzt genau eine Kundenmaske.** **2. Das System verpflichtet jeden Sachbearbeiter den Vor- und Zunamen, die Strasse, die PLZ, den Wohnort und das Geburtsdatum des Kunden in je einem Eingabefeld in der Kundenmaske zu erfassen.**
Nr. 16	Iteration 2
F	6T: Prüfen Sie Modaloperatoren der Notwendigkeit auf benötigte Ergänzungen Wie wird verpflichtet? → Es wird nicht verpflichtet, da der CRS-Server die Daten auf Vollständigkeit und Korrektheit überprüft. Siehe Anforderung 24. 3T: Decken Sie unvollständig spezifizierte Prozesswörter auf Wonach werden die Daten erfasst? → Die Daten werden nach dem Ausfüllen der Buchungsmaske erfasst. Siehe Anforderung 29.
E	**1. Das System besitzt genau eine Kundenmaske.** **2. Das System ermöglicht jedem Sachbearbeiter den Vor- und Zunamen, die Strasse, die PLZ, den Wohnort und das Geburtsdatum des Kunden in je einem Eingabefeld in der Kundenmaske zu erfassen.**

Nr. 17	Iteration 1
O	**Die Daten des Reiseanmelders - wie die Adresse oder das Alter - können aus einer Datenbank herausgeholt werden.**
U	1T: Formulieren Sie jede Anforderung im Aktiv Die Anforderung befindet sich im Passiv. 14G: Machen Sie konkrete Angaben über die Menge an Objekten Die Menge der Datenbanken ist nicht konkret. Ergänzung: „genau eine‟. 21W: Überprüfen Sie Nebensätze „Wie die Adresse oder das Alter‟ stellt keine Anforderung dar.
E	**Das System ermöglicht es die Daten des Reiseanmelders aus der Datenbank herauszuholen.** **Kommentar: Dies sind z.B. Adresse oder Alter**
F	3T: Decken Sie unvollständig spezifizierte Prozesswörter auf 5T: Klären Sie Mögliches und Unmögliches Wem wird das ermöglicht? Wie? Wann? → Dem Sachbearbeiter im Reisebüro. → Aus der Kundendatenbank beim Ausfüllen der Kundenmaske. 7T: Finden Sie implizite Annahmen Gibt es eine Datenbank? → Es gibt eine Kunden-Datenbank. Diese enthält alle wichtigen Informationen zu den Kunden (außer den Kontodaten). → Neueinträge erfolgen automatisch. Siehe Anforderung 21. 12G: Hinterfragen Sie Substantive ohne Bezugsindex Welche Daten genau? Von wem? Wie sehen sie aus? → Vorname, Name, Strasse, PLZ, Ort, Geburtsdatum des Reiseanmelders. 23W: Definieren Sie Substantive nach folgendem Schema Definiere Reiseanmelder! → Siehe Glossar.
E	**1. Das System besitzt genau eine Kundensuchmaske.** **2. Das System ermöglicht jedem Sachbearbeiter über genau einen Button „Kundensuche“ in der Kundenmaske die Kundensuchmaske zu öffnen.** **3. Das System ermöglicht jedem Sachbearbeiter über genau ein Eingabefeld für den Zunamen in der Kundensuchmaske nach allen Kunden mit dem gleichen Zunamen in der Kundendatenbank zu suchen.** **4. Wenn der Sachbearbeiter auf den Button „Suche starten“ in der Kundensuchmaske klickt, sucht das System in der Kundendatenbank nach allen Kunden mit dem gleichen Zunamen und stellt diese Kunden in genau einer Liste in der Kundensuchmaske dar.** **5. Die Liste enthält Vorname, Zuname, Strasse, PLZ, Wohnort**

	und Geburtsdatum jedes gefundenen Kunden. **6. Das System ermöglicht jedem Sachbearbeiter den Vor- und Zunamen, die Strasse, die PLZ, den Wohnort und das Geburtsdatum eines Kunden in der Liste in der Kundensuchmaske durch Doppelklick auf den Listeneintrag genau eines Kunden in die entsprechenden Eingabefelder der Kundenmaske zu übernehmen.** **7. Das System ermöglicht jedem Sachbearbeiter über den Button „Schließen“ die Kundensuchmaske zu verlassen.**

Nr. 18	Iteration 1
O	**Der Reiseanmelder ist auch derjenige, der später die Rechnung bezahlt.**
U	
E	
F	Hier handelt es sich um keine Anforderung. Es muss geklärt werden, ob eine Anforderung daraus entsteht oder ob es ein Kommentar oder Lexikoneintrag ist.
E	**→ Anforderung 18 wird als Glossareintrag umgewandelt.**

Nr. 19	Iteration 1
O	**Es müssen dann noch die Reisenden angegeben werden.**
U	1T: Formulieren Sie jede Anforderung im Aktiv Die Anforderung befindet sich im Passiv. 14G: Machen Sie konkrete Angaben über die Menge an Objekten Die Menge der Reisenden wird nicht genauer angegeben. Ergänzung: „jeden‟. 13G: Verwenden Sie Substantive stets in der Einzahl Reisenden steht im Plural. 12G: Hinterfragen Sie Substantive ohne Bezugsindex Die Reisenden sind nicht genauer spezifiziert. Ergänzung: „an der Reise teilnehmende‟. 20W: Vermeiden Sie Floskeln „Noch‟ ist überflüssig.
E	**Das System muss es ermöglichen jeden an der Reise teilnehmenden Reisenden anzugeben.**
F	3T: Decken Sie unvollständig spezifizierte Prozesswörter auf 5T: Klären Sie Mögliches und Unmögliches Wem muss es das ermöglichen? Wie wird angegeben? → Dem Sachbearbeiter. → Durch Angabe des Vor- und Nachnamens, Alters und Geschlecht (Mann, Frau, Kind, Baby) eines jeden Reisenden. 11G: Ermitteln Sie unvollständig spezifizierte Bedingungen Was ist wenn keine Reisenden vorhanden sind? Der Reiseanmeldende ein Reisender ist? → Der Reiseanmelder muss auch dann als Reisender erfasst werden. 14G: Machen Sie konkrete Angaben über die Menge an Objekten Wie viele Reisende gibt es? → Mindestens 1 Reisender. → Maximal 6 Reisende. 23W: Definieren Sie Substantive nach folgendem Schema Definiere Reisende! → Siehe Glossar.
E	**1. Das System ermöglicht jedem Sachbearbeiter über je zwei Eingabefelder pro Reisendem den Vor- und Zuname und das Alter und über genau ein Auswahlfeld das Geschlecht jedes Reisenden anzugeben.** **2. In allen Auswahlfeldern für das Geschlecht jedes Reisenden kann der Sachbearbeiter zwischen „Mann“, „Frau“, „Kind“ und „Baby“ wählen.** **3. Das System muss prüfen, ob für jeden Reisenden alle drei Eingabefelder gefüllt wurden.** **Kommentar: Nimmt der Kunde an der Reise teil, müssen dessen Name, Vorname, Alter und Geschlecht erneut angegeben wer-**

	den. Die Anzahl der an einer Reise teilnehmenden Erwachsenen und Kinder wurde bei der Suche der Reise bereits festgelegt.
Nr. 19	Iteration 2
F	6T: Prüfen Sie Modaloperatoren der Notwendigkeit auf benötigte Ergänzungen Was passiert wenn nicht alle angegeben wurden? → Der CRS-Server prüft die Daten auf Vollständigkeit und Korrektheit und gibt im Fehlerfall eine Fehlermeldung zurück. Siehe Anforderung 24.
E	**1. Das System ermöglicht jedem Sachbearbeiter über je zwei Eingabefelder pro Reisendem den Vor- und Zuname und das Alter und über genau ein Auswahlfeld das Geschlecht jedes Reisenden anzugeben.** **2. In allen Auswahlfeldern für das Geschlecht jedes Reisenden kann der Sachbearbeiter zwischen „Mann", „Frau", „Kind" und „Baby" wählen.**

Nr. 20	Iteration 1
O	**Dabei muss dann der Name, das Geschlecht und bei Kindern das Alter angegeben werden.**
U	1T: Formulieren Sie jede Anforderung im Aktiv Die Anforderung befindet sich im Passiv. 12G: Hinterfragen Sie Substantive ohne Bezugsindex Es ist nicht angegeben wessen Name etc. Ergänzung: „der Reisenden". 13G: Verwenden Sie Substantive stets in der Einzahl „Kinder" steht im Plural. 14G: Machen Sie konkrete Angaben über die Menge an Objekten Es wird keine Angabe über die Menge der Personen gemacht.
E	**Das System muss es ermöglichen den Namen und das Geschlecht jedes Reisenden und bei jedem mitreisenden Kind das Alter anzugeben.**
F	3T: Decken Sie unvollständig spezifizierte Prozesswörter auf 5T: Klären Sie Mögliches und Unmögliches Wem wird das ermöglicht? Wie wird angegeben? → Dem Sachbearbeiter im Reisebüro. → Über Felder in einer Eingabemaske. → Es wurde erkannt, dass auch bei Erwachsenen das Alter in diesem Schritt der Reisbuchung angegeben werden muss.
E	**Anforderung 20 wurde mit Anforderung 19 verschmolzen.**

Nr. 21	Iteration 1
O	**Die Kontodaten des Kunden dürfen aus rechtlichen Gründen nicht mitgespeichert werden. .**
U	1T: Formulieren Sie jede Anforderung im Aktiv Die Anforderung befindet sich im Passiv. 21W: Überprüfen Sie Nebensätze „Aus rechtlichen Gründen" stellt nur eine Begründung dar.
E	**Das System darf die Kontodaten des Kunden nicht mitspeichern.** **Kommentar: Aus rechtlichen Gründen.**
F	5T: Klären Sie Mögliches und Unmögliches Wann werden Daten gespeichert? Wie genau? → Beim Absenden der Buchung, wenn kein gleicher Kunde vorhanden ist, zudem Anlage einer Reiseliste 7T: Finden Sie implizite Annahmen Werden Daten gespeichert? Es werden Kontodaten eingegeben. → Ja, beim Abschicken der Buchung werden die Kundendaten und die Reisedaten gespeichert. → Kontodaten werden nach Auswahl der Zahlungsart eingegeben. Siehe Anforderung 29. 12G: Hinterfragen Sie Substantive ohne Bezugsindex Welches Kunden? → Die Daten des Reiseanmelders werden gespeichert. 11G: Ermitteln Sie unvollständig spezifizierte Bedingungen Was ist, wenn keine Kontodaten vorhanden sind? → Dann wurde eine andere Zahlungsart vereinbart, zum Beispiel Zahlung per Rechung (hier werden nur die Adressdaten benötigt) oder Zahlung per Kreditkarte (hier werden die Kreditkartendaten benötigt). 23W: Definieren Sie Substantive nach folgendem Schema Definiere Kontodaten! → Wurde genauer in die Anforderung übernommen.
E	**1. Wenn der Sachbearbeiter die Buchung genau einer Pauschalreise absendet und kein Kunde mit sowohl gleichem Vorname und Zunamen und Strasse und PLZ und Ort und Geburtsdatum in der Kundendatenbank vorhanden ist, speichert das System Vorname, Zuname, Strasse, PLZ, Ort und Geburtsdatum in der Kundendatenbank ab und legt genau eine Reiseliste für den Kunde in der Kundendatenbank an.** **2. Wenn der Sachbearbeiter die Buchung genau einer Pauschalreise absendet , fügt das System das Reiseziel, den Reiseort, das Hotel, den Preis und die Anzahl der Reisenden der zu buchenden Pauschalreise der Reiseliste des Kunde in der Kundendatenbank hinzu.** **3. Das System darf die Kontodaten des Kunden nicht mitspeichern.** **Kommentar: Aus rechtlichen Gründen.**

Eingrenzung des Systems:
Durch das Regelwerk wurde erkannt, das implizit der Aufbau einer Kundendatenbank gewünscht ist. Aus Gründen der Komplexitätsreduktion wird nur die Speicherung von Kundendaten behandelt. Auf weitere notwendige Pflegetransaktionen wird nicht eingegangen.

Nr. 22	Iteration 1
O	**Der Kunde kann dann noch eine Zahlungsart auswählen.**
U	3T: Decken Sie unvollständig spezifizierte Prozesswörter auf Es ist nicht angegeben wer die Auswahl ermöglicht. Ergänzung: „Das System‟. 14G: Machen Sie konkrete Angaben über die Menge an Objekten Es wird keine definierte Angabe für Menge der Zahlungsarten gemacht. Ergänzung: „genau eine‟. 20W: Vermeiden Sie Floskeln „Noch‟ ist überflüssig.
E	**Das System ermöglicht es jedem Kunden genau eine Zahlungsart auszuwählen.**
F	5T: Klären Sie Mögliches und Unmögliches Wie wird ausgewählt? Woraus? → Wird durch den Sachbearbeiter im Reisebüro über eine Auswahlliste ausgewählt. → Der Sachbearbeiter erledigt dies. 7T: Finden Sie implizite Annahmen Gibt es eine Liste mit Zahlungsarten? → Es gibt drei Möglichkeiten. 12G: Hinterfragen Sie Substantive ohne Bezugsindex Welcher Kunde? → Der Reiseanmelder. 14G: Machen Sie konkrete Angaben über die Menge an Objekten Wie viele Zahlungsarten gibt es? → Wurde aus Anforderung 23 übernommen.
E	**1. Das System ermöglicht jedem Sachbearbeiter in der Kundenmaske über genau eine Auswahlliste als Zahlungsart entweder „Zahlung per Lastschrift“ oder „Zahlung per Rechnung“ oder „Zahlung per Kreditkarte“ auszuwählen.** **2. Wenn der Sachbearbeiter „Zahlung per Rechnung“ auswählt, stellt das System keine weiteren Eingabefelder bereit.** **3. Wenn der Sachbearbeiter „Zahlung per Lastschrift“ auswählt, stellt das System dem Sachbearbeiter je genau ein verpflichtendes Eingabefeld für die Kontonummer, die Bankleitzahl und den Kontoinhaber in der Kundenmaske bereit.**

	4. Wenn der Sachbearbeiter „Zahlung per Kreditkarte“ auswählt, stellt das System dem Sachbearbeiter je genau ein verpflichtendes Eingabefeld für die Kreditkartennummer, die Kreditkartenfirma, die Kreditkartengültigkeit und die Kreditkartenprüfnummer in der Kundenmaske bereit.
Nr. 22	Iteration 2
F	6T: Prüfen Sie Modaloperatoren der Notwendigkeit auf benötigte Ergänzungen Was ist wenn keine Zahlungsdaten eingegeben werden. → Der CRS-Server prüft die Daten auf Vollständigkeit und Korrektheit und gibt im Fehlerfall eine Fehlermeldung zurück. Siehe Anforderung 24.
E	**1. Das System ermöglicht jedem Sachbearbeiter in der Kundenmaske über genau eine Auswahlliste als Zahlungsart entweder „Zahlung per Lastschrift“ oder „Zahlung per Rechnung“ oder „Zahlung per Kreditkarte“ auszuwählen.** **2. Wenn der Sachbearbeiter „Zahlung per Rechnung“ auswählt, stellt das System keine weiteren Eingabefelder bereit.** **3. Wenn der Sachbearbeiter „Zahlung per Lastschrift“ auswählt, stellt das System dem Sachbearbeiter je genau ein verpflichtendes Eingabefeld für die Kontonummer, die Bankleitzahl und den Kontoinhaber in der Kundenmaske bereit.** **4. Wenn der Sachbearbeiter „Zahlung per Kreditkarte“ auswählt, stellt das System dem Sachbearbeiter je genau ein verpflichtendes Eingabefeld für die Kreditkartennummer, die Kreditkartenfirma, die Kreditkartengültigkeit und die Kreditkartenprüfnummer in der Kundenmaske bereit.**

Hier wurde vom Anforderungssteller bemerkt, dass dieser Schritt nach Anforderung 20 erfolgen muss, da sonst der zeitliche Ablauf Brüche aufweist. Erst nach der Auswahl der Zahlungsart werden die Kontodaten eingegeben.

Nr. 23	Iteration 1
O	**Als Zahlungsarten gibt es Zahlung per Lastschrift, mit Kreditkarte oder per Rechnung.**
U	
E	
F	Hier handelt sich um keine Anforderung. Es muss geklärt werden, ob eine Anforderung daraus entsteht oder ob es ein Kommentar oder Lexikoneintrag ist.
E	**Anforderung 23 wurde mit Anforderung 22 verschmolzen und als Glossareintrag aufgenommen.**

Nr. 24	Iteration 1
O	**Danach kann die Buchung abgeschickt werden und geht dann zum Reiseveranstalter über.**
U	1T: Formulieren Sie jede Anforderung im Aktiv Die Anforderung befindet sich im Passiv. 14G: Machen Sie konkrete Angaben über die Menge an Objekten Die Menge der Reiseveranstalter ist nicht spezifiziert. Ergänzung: „genau einer“. 22W: Stellen Sie zeitliche Abhängigkeiten klar Es wird eine Abfolge dargestellt.
E	**Wenn die Buchung genau einer Reise durch das System abgeschickt wird, geht die Buchung genau an einen Reiseveranstalter über.**
F	3T: Decken Sie unvollständig spezifizierte Prozesswörter auf 5T: Klären Sie Mögliches und Unmögliches Wohin wird die Buchung geschickt? Wer ist noch beteiligt? Wie wird abgeschickt? → Der Sachbearbeiter, das Reisebuchungssystem und der CRS-Server. → Durch einen Button. → Hier wurde durch Rücksprache mit dem Anforderungssteller geklärt, dass die Buchung an das CRS-System übergeht. Von hieraus geht die Buchung dann zu dem jeweiligen Reiseveranstalter über. 6T: Prüfen Sie Modaloperatoren der Notwendigkeit auf benötigte Ergänzungen Was ist wenn die Buchung nicht übergeht? → Kann die Buchung nicht an das CRS-System abgeschickt werden oder treten beim Senden Fehler auf, so soll eine Fehlermeldung ausgegeben werden und die Daten bestehen bleiben. → Konnte die Buchung erfolgreich übermittelt werden, so soll eine Erfolgsmeldung ausgegeben werden. 7T: Finden Sie implizite Annahmen Gibt es eine Liste mit Reiseveranstaltern? → Kann gestrichen werden. 8T: Überprüfen Sie das Verhältnis der Objekte

	Wie stehen Buchung und Reiseveranstalter zueinander? → Jede Buchung bezieht sich auf einen Veranstalter. Die Information besitzt der CRS-Server. 12G: Hinterfragen Sie Substantive ohne Bezugsindex Welche Buchung? Was genau? → Siehe Anmerkung im Nachfolgenden.
E	**1. Wenn der Sachbearbeiter auf den Button „Buchung absenden" in der Kundenmaske klickt, sendet das System die Buchungsdaten an den CRS-Server, wartet auf die Erfolgsmeldung des CRS-Servers und gibt dem Sachbearbeiter den Hinweis „Buchung erfolgt" aus.** **2. Wenn das System die Buchungsdaten an den CRS-Server sendet und innerhalb von 10 Sekunden keine Erfolgsmeldung oder eine Fehlermeldung vom CRS-Server erhält, soll das System dem Sachbearbeiter den Hinweis „Buchung nicht erfolgt" ausgeben und ermöglichen die Buchung erneut abzusenden.**
Nr. 24	Iteration 2
F	In der Validation wurde erkannt: → Aus den Anforderungen 16, 19 und 22 ergab sich, dass auch Fehlermeldung vom CRS-Server an den Sachbearbeiter weitergeleitet werden sollten.
E	**1. Wenn der Sachbearbeiter auf den Button „Buchung absenden" in der Kundenmaske klickt, sendet das System die Buchungsdaten an den CRS-Server, wartet auf die Erfolgsmeldung des CRS-Servers und gibt dem Sachbearbeiter den Hinweis „Buchung erfolgt" aus.** **2. Wenn das System die Buchungsdaten an den CRS-Server sendet und innerhalb von 10 Sekunden keine Erfolgsmeldung vom CRS-Server erhält, soll das System dem Sachbearbeiter den Hinweis „Buchung nicht erfolgt" ausgeben und ermöglichen die Buchung erneut abzusenden.** **3. Wenn das System die Buchungsdaten an den CRS-Server sendet und genau eine Fehlermeldung vom CRS-Server erhält, soll das System dem Sachbearbeiter die Fehlermeldung ausgeben und in der Buchungsmaske verbleiben.**

Eingrenzung des Systems:

Das Regelwerk würde eine genauere Spezifikation der „Buchungsdaten" bedingen. Aus Gründen der Komplexitätsreduktion wird auf die genaue Beschreibung verzichtet. Ebenso endet die Anforderungserhebung an dieser Stelle und es werden keine weiteren Schritte spezifiziert.

Nr. 25 – (neu seit Iteration 1)	Iteration 1
O	**Die Diskussion mit dem Anforderungssteller in der ersten Iteration ergab, dass die Werte für die Auswahllisten in der Suchmaske vom CRS-Server abgerufen werden. Aus den Angaben des Anforderungsstellers wurde die Anforderung vom Projektteam selbst spezifiziert. Es ergab sich dabei, dass auch das Verhalten spezifiziert werden muss, für den Fall, dass die Daten zum Beispiel aufgrund einer fehlenden Verbindung nicht vom CRS-Server abgefragt werden können.**
U	
E	**1. Wenn der Sachbearbeiter die Suchmaske öffnet, muss das System immer alle vorhandenen Werte für die Einträge in den Auswahllisten für den Abflugs- und Ankunftsflughafen, die Reisekategorie, die Reiseart, das Reiseziel, den Reiseort, die Hotelkategorie und die Verpflegungsart vom CRS-Server abrufen.** **2. Wenn der Sachbearbeiter in der Einstiegsmaske die Suche nach Pauschalreisen ausgewählt hat und das System nicht innerhalb von 5 Sekunden alle vorhandenen Werte für die Einträge in den Auswahllisten für den Abflugs- und Ankunftsflughafen, die Reisekategorie, die Reiseart, das Reiseziel, den Reiseort, die Hotelkategorie und die Verpflegungsart vom CRS-Server abrufen kann, so soll das System dem Sachbearbeiter den Hinweis „CRS-Server nicht erreichbar" ausgeben und in der Einstiegsmaske verbleiben.**
F	
E	**1. Wenn der Sachbearbeiter die Suchmaske öffnet, muss das System immer alle vorhandenen Werte für die Einträge in den Auswahllisten für den Abflugs- und Ankunftsflughafen, die Reisekategorie, die Reiseart, das Reiseziel, den Reiseort, die Hotelkategorie und die Verpflegungsart vom CRS-Server abrufen.** **2. Wenn der Sachbearbeiter in der Einstiegsmaske die Suche nach Pauschalreisen ausgewählt hat und das System nicht innerhalb von 5 Sekunden alle vorhandenen Werte für die Einträge in den Auswahllisten für den Abflugs- und Ankunftsflughafen, die Reisekategorie, die Reiseart, das Reiseziel, den Reiseort, die Hotelkategorie und die Verpflegungsart vom CRS-Server abrufen kann, so soll das System dem Sachbearbeiter den Hinweis „CRS-Server nicht erreichbar" ausgeben und in der Einstiegsmaske verbleiben.**

Nr. 26 – (neu seit Iteration 1)	Iteration 1
O	**Die Diskussion mit dem Anforderungssteller in der ersten Iteration ergab, dass die Informationen für Detailanzeige vom CRS-Server abgerufen werden müssen. Auch hier musste das Verhalten im Fehlerfall spezifiziert werden.**
U	
E	**1. Wenn der Sachbearbeiter die Detailanzeige zu genau einer Pauschalreise in der Ergebnisliste öffnet, ruft das System immer alle vorhandenen Detailinformationen des Hotels zu der gewählten Pauschalreise vom CRS-Server ab und stellt sie im Fließtext dar.** **2. Wenn keine Detailinformationen zu dem Hotel der gewählten Pauschalreise auf dem CRS-Server vorhanden sind, soll das System den Hinweis „Keine Detailinformationen vorhanden“ ausgeben und in der Ergebnisliste verbleiben.** **3. Wenn der Sachbearbeiter die Detailanzeige zu dem Hotel der gewählten Pauschalreise in der Ergebnisliste öffnet und das System die Detailinformationen des Hotels vom CRS-Server nicht innerhalb von 5 Sekunden abrufen kann, soll es dem Sachbearbeiter den Hinweis „CRS-Server nicht erreichbar“ ausgeben und in der Ergebnisliste verbleiben.**
F	
E	**1. Wenn der Sachbearbeiter die Detailanzeige zu genau einer Pauschalreise in der Ergebnisliste öffnet, ruft das System immer alle vorhandenen Detailinformationen des Hotels zu der gewählten Pauschalreise vom CRS-Server ab und stellt sie im Fließtext dar.** **2. Wenn keine Detailinformationen zu dem Hotel der gewählten Pauschalreise auf dem CRS-Server vorhanden sind, soll das System den Hinweis „Keine Detailinformationen vorhanden“ ausgeben und in der Ergebnisliste verbleiben.** **3. Wenn der Sachbearbeiter die Detailanzeige zu dem Hotel der gewählten Pauschalreise in der Ergebnisliste öffnet und das System die Detailinformationen des Hotels vom CRS-Server nicht innerhalb von 5 Sekunden abrufen kann, soll es dem Sachbearbeiter den Hinweis „CRS-Server nicht erreichbar“ ausgeben und in der Ergebnisliste verbleiben.**

Nr. 27 – (neu seit Iteration 1)	Iteration 1
O	**Die Diskussion mit dem Anforderungssteller in der ersten Iteration ergab, dass auch eine Auswahl an Flügen bei der Buchung der Reise notwendig ist.**
U	
E	**Das System verpflichtet den Sachbearbeiter über je eine Auswahlliste in der Buchungsmaske genau einen Hinflug und genau einen Rückflug zu genau einer gewählten Pauschalreise auszuwählen.**
F	6T: Prüfen Sie Modaloperatoren der Notwendigkeit auf benötigte Ergänzungen Wie wird der Sachbearbeiter verpflichtet? → Bei der Weiter-Navigation tritt eine Fehlermeldung auf. Siehe Anforderung 29.
E	**Das System ermöglicht dem Sachbearbeiter über je eine Auswahlliste in der Buchungsmaske genau einen Hinflug und genau einen Rückflug zu genau einer gewählten Pauschalreise auszuwählen.**

Nr. 28 – (neu seit Iteration 1)	Iteration 1
O	**Die Diskussion mit dem Anforderungssteller in der ersten Iteration ergab, dass die Auswahl an Flügen vom CRS Server abgerufen werden muss. Auch hier musste das Verhalten im Fehlerfall spezifiziert werden.**
U	
E	**1. Wenn der Sachbearbeiter die Buchungsmaske öffnet, ruft das System immer alle auf dem CRS-Server verfügbaren Zusatzoptionen und alle verfügbaren Flüge der gewählten Pauschalreise vom CRS-Server ab.** **2. Wenn der Sachbearbeiter die Buchungsmaske öffnet und das System nicht innerhalb von 5 Sekunden vom CRS-Server alle Zusatzoptionen genau einer Pauschalreise abrufen kann, soll das System dem Sachbearbeiter den Hinweis „CRS-Server nicht erreichbar“ ausgeben und in der Ergebnisliste verbleiben.**
F	
E	**1. Wenn der Sachbearbeiter die Buchungsmaske öffnet, ruft das System immer alle auf dem CRS-Server verfügbaren Zusatzoptionen und alle verfügbaren Flüge der gewählten Pauschalreise vom CRS-Server ab.** **2. Wenn der Sachbearbeiter die Buchungsmaske öffnet und das System nicht innerhalb von 5 Sekunden vom CRS-Server alle Zusatzoptionen genau einer Pauschalreise abrufen kann, soll das System dem Sachbearbeiter den Hinweis „CRS-Server nicht erreichbar“ ausgeben und in der Ergebnisliste verbleiben.**

Nr. 29 – (neu seit Iteration 2)	Iteration 1
O	**Die Diskussion mit dem Anforderungssteller in der zweiten Iteration ergab, dass von der Buchungsmaske zur Kundenmaske und zurück zur Ergebnisliste gesprungen werden können muss. Des Weiteren muss beim Übergang zur Kundenmaske eine Fehlermeldung ausgegeben werden, falls keine Flüge ausgewählt wurden.**
U	
E	**1. Das System ermöglicht dem Sachbearbeiter über einen Button „Weiter“ in der Buchungsmaske zur Kundenmaske zu wechseln.** **2. Das System ermöglicht dem Sachbearbeiter über einen Button „Zurück“ in der Buchungsmaske zur Ergebnisliste zu wechseln.** **3. Wenn der Sachbearbeiter mindestens den Hinflug oder den Rückflug in der Buchungsmaske nicht ausgewählt hat, soll das System bei Klick auf den Button „Weiter“ in der Buchungsmaske dem Sachbearbeiter den Hinweis „Flugauswahl fehlerhaft“ ausgeben und in der Buchungsmaske verbleiben.**
F	
E	**1. Das System ermöglicht dem Sachbearbeiter über einen Button „Weiter“ in der Buchungsmaske zur Kundenmaske zu wechseln.** **2. Das System ermöglicht dem Sachbearbeiter über einen Button „Zurück“ in der Buchungsmaske zur Ergebnisliste zu wechseln.** **3. Wenn der Sachbearbeiter mindestens den Hinflug oder den Rückflug in der Buchungsmaske nicht ausgewählt hat, soll das System bei Klick auf den Button „Weiter“ in der Buchungsmaske dem Sachbearbeiter den Hinweis „Flugauswahl fehlerhaft“ ausgeben und in der Buchungsmaske verbleiben.**

3.6 Ergebnis der Anwendung

Nachfolgend werden die sich ergebenden Anforderungen und Glossareinträge zusammenfassend dargestellt, wobei eine Sortierung nach zeitlichen und thematischen Gesichtspunkten vorgenommen wird.

3.6.1 Anforderungen

1. Das System besitzt genau eine Einstiegsmaske.

2. Das System ermöglicht jedem Sachbearbeiter im Reisebüro über je ein Ankreuzfeld auf der Einstiegsmaske zwischen der Suche nach entweder Hotels oder Flügen oder Pauschalreisen zu wählen.
   Kommentar: Je nach Wunsch des Kunden.

   Reduzierung des Umfangs: Die weiteren Anforderungen beschränken sich auf die Leistung „Pauschalreise". Anforderungen für das Buchen von Flügen oder einzelnen Hotelaufenthalten werden nicht berücksichtigt.

3. Das System soll jedem Sachbearbeiter genau eine Suchmaske für die Eingabe aller Suchkriterien von Pauschalreisen bereitstellen.

4. Wenn der Sachbearbeiter die Suchmaske öffnet, muss das System immer alle vorhandenen Werte für die Einträge in den Auswahllisten für den Abflugs- und Ankunftsflughafen, die Reisekategorie, die Reiseart, das Reiseziel, den Reiseort, die Hotelkategorie und die Verpflegungsart vom CRS-Server abrufen.

   Reduzierung des Umfangs: Auf die genauere Beschreibung der Kommunikation und der Vernetzung des Systems mit dem CRS-Server wird hier aus Komplexitätsgründen verzichtet.

5. Wenn der Sachbearbeiter in der Einstiegsmaske die Suche nach Pauschalreisen ausgewählt hat und das System nicht innerhalb von 5 Sekunden alle vorhandenen Werte für die Einträge in den Auswahllisten für den Abflugs- und Ankunftsflughafen, die Reisekategorie, die Reiseart, das Reiseziel, den Reiseort, die Hotelkategorie und die Verpflegungsart vom CRS-Server abrufen kann, so soll das System dem Sachbearbeiter den Hinweis „CRS-Server nicht erreichbar" ausgeben und in der Einstiegsmaske verbleiben.

6. Das System ermöglicht jedem Sachbearbeiter über je ein Datumseingabefeld in der Suchmaske den frühesten Start- und den spätesten Endtermin für die gesuchten Pauschalreisen festzulegen.

7. Das System belegt das Eingabefeld für den Starttermin mit dem heutigen Datum plus 3 Tage und das Eingabefeld für den Endtermin mit dem heutigen Datum plus 24 Tage.

8. Das System ermöglicht jedem Sachbearbeiter über je genau ein Zahleingabefeld in der Suchmaske die minimale Reisedauer und die maximale Reisedauer der gesuchten Pauschalreisen in Tagen einzugeben.

9. Das System belegt das Zahleingabefeld für die minimale Reisedauer mit 0 und das Zahleingabefeld für die maximale Reisedauer mit 7.

10. Das System ermöglicht jedem Sachbearbeiter über genau ein Auswahlfeld in der Suchmaske die Gesamtanzahl der an den gesuchten Pauschalreisen reisenden Personen festzulegen.
Kommentar: Die reisenden Personen umfassen die reisenden Erwachsenen und die reisenden Kinder.

11. Die Anzahl der reisenden Personen muss zwischen 1 und 6 liegen.

12. Das System belegt das Auswahlfeld für die Gesamtanzahl der reisenden Personen mit 2.

13. Das System ermöglicht jedem Sachbearbeiter das Alter jedes reisenden Kindes über genau eine der drei Auswahllisten anzugeben.

14. Jede Auswahlliste für das Alter der reisenden Kinder enthält die Werte von 0 bis 16 und den Wert „–".

15. Das System belegt jede Auswahlliste für das Alter der reisenden Kinder mit dem Wert „-" vor.

16. Das System berechnet die Anzahl der reisenden Kinder aus der Anzahl der nicht mit „-" belegten Auswahllisten für das Alter der reisenden Kinder.

17. Das System berechnet die Anzahl der reisenden Erwachsenen aus der Gesamtanzahl der reisenden Personen minus der Anzahl der reisenden Kinder.

18. Das System ermöglicht jedem Sachbearbeiter je null bis alle Einträge aus je genau einer Auswahlliste für den Abflugs- und Ankunftsflughafen, die Reisekategorie, die Reiseart, das Reiseziel und den Reiseort der zu suchenden Pauschalreise auszuwählen.

19. Das System ermöglicht jedem Sachbearbeiter über genau eine Auswahlliste in der Suchmaske optional genau eine Hotelkategorie für alle zu suchenden Pauschalreisen auszuwählen.

20. Das System ermöglicht jedem Sachbearbeiter über genau eine Auswahlliste keine bis alle Verpflegungsarten für alle zu suchenden Pauschalreisen auszuwählen.

21. Das System ermöglicht jedem Sachbearbeiter über zwei Zahleingabefelder in der Suchmaske optional die Untergrenze und optional die Obergrenze für den Preis aller zu suchender Pauschalreisen anzugeben.

22. Wenn der Sachbearbeiter die Suchkriterien des Kunden für alle zu suchenden Pauschalreisen in die Suchmaske eingegeben hat und auf den Button „Suche starten“ in der Suchmaske gedrückt hat, schickt das System die Suchanfrage an den CRS-Server ab und stellt das vom CRS-Server zurückgelieferte Suchergebnis in genau einer Ergebnisliste dar.
Reduzierung des Umfangs: Auf eine detaillierte Beschreibung der Ergebnisliste und der Vernetzung mit dem CRS-Server wird aufgrund der hohen Komplexität verzichtet.

23. Die Ergebnisliste soll alle auf dem CRS-Server verfügbaren Pauschalreisen
mit Reisezeitraum innerhalb dem ausgewählten frühesten Starttermin und spätesten Endtermin
und Preis innerhalb der ausgewählten Untergrenze und Obergrenze
und Reisedauer innerhalb der ausgewählten Unter- und Obergrenze
und ausgewählter Zahl der reisenden Erwachsenen und Kindern
und ausgewählter oder höherer als ausgewählter Hotelkategorie
und einer der ausgewählten Verpflegungsarten
und einem der ausgewählten Abflugshäfen
und einem der ausgewählten Ankunftsflughäfen
und einem der ausgewählten Reiseorte
und einem der ausgewählten Reiseziele
und einer der ausgewählten Reisekategorien
und einer der ausgewählten Reisearten enthalten.

24. Wenn das Eingabefeld für die minimale Reisedauer keinen Wert enthält, soll die Ergebnisliste alle Pauschalreisen mit der Reisedauer gleich oder größer „0 Tage“ enthalten.

25. Wenn das Eingabefeld für die maximale Reisedauer keinen Wert enthält, soll die Ergebnisliste alle Pauschalreisen mit der Reisedauer kleiner oder gleich „7 Tage“ enthalten.

26. Wenn das Eingabefeld für den frühesten Starttermin keinen Wert enthält, soll die Ergebnisliste alle Pauschalreisen mit Starttermin größer oder gleich dem heutigen Datum plus 3 Tage enthalten.

27. Wenn das Eingabefeld für den spätesten Endtermin keinen Wert enthält, soll die Ergebnisliste alle Pauschalreisen mit Endtermin kleiner oder gleich dem heutigen Datum plus 24 Tage enthalten.

28. Wenn keine Hotelkategorie
oder keine Verpflegungsart
oder keine Untergrenze des Preises
oder keine Obergrenze des Preises
oder kein Ankunftsflughafen
oder kein Abflugsflughafen
oder kein Reiseort
oder kein Reiseziel
oder keine Reisekategorie
oder keine Reiseart

ausgewählt wurde, soll jeweils nach allen möglichen Werten jedes nicht gewählten Suchkriteriums gesucht werden.

29. Wenn die Ergebnisliste keine Ergebniseinträge zu den Suchkriterien enthält, gibt das System dem Sachbearbeiter den Hinweis „Keine Einträge gefunden" aus, verbleibt in der Suchmaske und ermöglicht dem Sachbearbeiter alle Suchkriterien in der Suchmaske zu ändern.

30. Wenn der Sachbearbeiter in der Suchmaske den Button „Suche starten" gedrückt hat und das System vom CRS nicht innerhalb von 10 Sekunden vom CRS-Server genau eine Ergebnisliste abrufen kann, soll das System dem Sachbearbeiter den Hinweis „CRS-Server nicht erreichbar" ausgeben und ermöglichen die Suche abzubrechen oder neu zu starten.

31. Das System ermöglicht es jedem Sachbearbeiter jedes vom Kunden gewünschte Hotel aus der Ergebnisliste über einen Doppelklick mit der linken Maustaste für genau eine Detailanzeige auszuwählen.

32. Das System ermöglicht jedem Sachbearbeiter über den Button „Zurück zur Suche" in der Ergebnisliste zur Suchmaske zurückzukehren.

33. Das System stellt jedem Sachbearbeiter genau eine Detailanzeige mit allen Detailinformationen zu jedem Hotel aus der Ergebnisliste für jeden Kunden bereit.

34. Wenn der Sachbearbeiter die Detailanzeige zu genau einer Pauschalreise in der Ergebnisliste öffnet, ruft das System immer alle vorhandenen Detailinformationen des Hotels zu der gewählten Pauschalreise vom CRS-Server ab und stellt sie im Fließtext dar.

    Reduzierung des Umfangs: Auf die genauere Beschreibung der Kommunikation und der Vernetzung des Systems mit dem CRS-Server wird hier aus Komplexitätsgründen verzichtet.

35. Wenn keine Detailinformationen zu dem Hotel der gewählten Pauschalreise auf dem CRS-Server vorhanden sind, soll das System den Hinweis „Keine Detailinformationen vorhanden" ausgeben und in der Ergebnisliste verbleiben.

36. Wenn der Sachbearbeiter die Detailanzeige zu dem Hotel der gewählten Pauschalreise in der Ergebnisliste öffnet und das System die Detailinformationen des Hotels vom CRS-Server nicht innerhalb von 5 Sekunden abrufen kann, soll es dem Sachbearbeiter den Hinweis „CRS-Server nicht erreichbar" ausgeben und in der Ergebnisliste verbleiben.

37. Jeder Eintrag in der Ergebnisliste der Pauschalreisen besitzt genau einen Button „zum Buchen".

38. Das System ermöglicht jedem Sachbearbeiter genau eine Pauschalreise für die Reisebuchung auszuwählen indem er auf den Button

„zum Buchen“ in dem Eintrag der Ergebnisliste der gewählten Pauschalreisen klickt.
Kommentar: Sobald der Kunde weiß, welche Pauschalreise aus der Ergebnisliste er will.

39. Das System besitzt genau eine Buchungsmaske.

40. Wenn der Sachbearbeiter die Buchungsmaske öffnet, ruft das System immer alle auf dem CRS-Server verfügbaren Zusatzoptionen und alle verfügbaren Flüge der gewählten Pauschalreise vom CRS-Server ab.

41. Wenn der Sachbearbeiter die Buchungsmaske öffnet und das System nicht innerhalb von 5 Sekunden vom CRS-Server alle Zusatzoptionen genau einer Pauschalreise abrufen kann, soll das System dem Sachbearbeiter den Hinweis „CRS-Server nicht erreichbar“ ausgeben und in der Ergebnisliste verbleiben.

42. Bei der Reisebuchung ermöglicht das System jedem Sachbearbeiter über je ein Zahleingabefeld in der Buchungsmaske genau eine Bestellmenge zu jeder Zusatzoption des Hotels und zu jeder Zusatzoption des Reiseveranstalters der gewählten Pauschalreise einzugeben.

43. Das System ermöglicht dem Sachbearbeiter über je eine Auswahlliste in der Buchungsmaske genau einen Hinflug und genau einen Rückflug zu genau einer gewählten Pauschalreise auszuwählen.

44. Das System ermöglicht jedem Sachbearbeiter optional bei der Reisebuchung weitere unverbindliche Wünsche des Kunden in genau einem 50 Zeichen langem Freitextfeld in der Buchungsmaske anzugeben.

45. Das System ermöglicht dem Sachbearbeiter über einen Button „Weiter“ in der Buchungsmaske zur Kundenmaske zu wechseln.

46. Das System ermöglicht dem Sachbearbeiter über einen Button „Zurück“ in der Buchungsmaske zur Ergebnisliste zu wechseln.

47. Wenn der Sachbearbeiter mindestens den Hinflug oder den Rückflug in der Buchungsmaske nicht ausgewählt hat, soll das System bei Klick auf den Button „Weiter“ in der Buchungsmaske dem Sachbearbeiter den Hinweis „Flugauswahl fehlerhaft“ ausgeben und in der Buchungsmaske verbleiben.

48. Das System besitzt genau eine Kundenmaske.

49. Das System ermöglicht jedem Sachbearbeiter den Vor- und Zunamen, die Strasse, die PLZ, den Wohnort und das Geburtsdatum des Kunden in je einem Eingabefeld in der Kundenmaske zu erfassen.

50. Das System besitzt genau eine Kundensuchmaske.

51. Das System ermöglicht jedem Sachbearbeiter über genau einen Button „Kundensuche“ in der Kundenmaske die Kundensuchmaske zu öffnen.

52. Das System ermöglicht jedem Sachbearbeiter über genau ein Eingabefeld für den Zunamen in der Kundensuchmaske nach allen Kunden mit dem gleichen Zunamen in der Kundendatenbank zu suchen.

53. Wenn der Sachbearbeiter auf den Button „Suche starten‟ in der Kundensuchmaske klickt, sucht das System in der Kundendatenbank nach allen Kunden mit dem gleichen Zunamen und stellt diese Kunden in genau einer Liste in der Kundensuchmaske dar.

54. Die Liste enthält Vorname, Zuname, Strasse, PLZ, Wohnort und Geburtsdatum jedes gefundenen Kunden.

55. Das System ermöglicht jedem Sachbearbeiter den Vor- und Zunamen, die Strasse, die PLZ, den Wohnort und das Geburtsdatum eines Kunden in der Liste in der Kundensuchmaske durch Doppelklick auf den Listeneintrag genau eines Kunden in die entsprechenden Eingabefelder der Kundenmaske zu übernehmen.

56. Das System ermöglicht jedem Sachbearbeiter über den Button „Schließen‟ die Kundensuchmaske zu verlassen.

57. Das System ermöglicht jedem Sachbearbeiter über je zwei Eingabefelder pro Reisendem den Vor- und Zuname und das Alter und über genau ein Auswahlfeld das Geschlecht jedes Reisenden anzugeben.
Kommentar: Nimmt der Kunde an der Reise teil, müssen dessen Name, Vorname, Alter und Geschlecht erneut angegeben werden. Die Anzahl der an einer Reise teilnehmenden Erwachsenen und Kinder wurde bei der Suche der Reise bereits festgelegt.

58. In allen Auswahlfeldern für das Geschlecht jedes Reisenden kann der Sachbearbeiter zwischen „Mann‟, „Frau‟, „Kind‟ und „Baby‟ wählen.

59. Das System ermöglicht jedem Sachbearbeiter in der Kundenmaske über genau eine Auswahlliste als Zahlungsart entweder „Zahlung per Lastschrift‟ oder „Zahlung per Rechnung‟ oder „Zahlung per Kreditkarte‟ auszuwählen.

60. Wenn der Sachbearbeiter „Zahlung per Rechnung‟ auswählt, stellt das System keine weiteren Eingabefelder bereit.

61. Wenn der Sachbearbeiter „Zahlung per Lastschrift‟ auswählt, stellt das System dem Sachbearbeiter je genau ein verpflichtendes Eingabefeld für die Kontonummer, die Bankleitzahl und den Kontoinhaber in der Kundenmaske bereit.

62. Wenn der Sachbearbeiter „Zahlung per Kreditkarte‟ auswählt, stellt das System dem Sachbearbeiter je genau ein verpflichtendes Eingabefeld für die Kreditkartennummer, die Kreditkartenfirma, die Kreditkartengültigkeit und die Kreditkartenprüfnummer in der Kundenmaske bereit.

63. Wenn der Sachbearbeiter auf den Button „Buchung absenden" in der Kundenmaske klickt, sendet das System die Buchungsdaten an den CRS-Server, wartet auf die Erfolgsmeldung des CRS-Servers und gibt dem Sachbearbeiter den Hinweis „Buchung erfolgt" aus.

    Eingrenzung des Systems: Das Regelwerk würde eine genauere Spezifikation der „Buchungsdaten" bedingen. Aus Gründen der Komplexitätsreduktion wird auf die genaue Beschreibung verzichtet.

64. Wenn der Sachbearbeiter die Buchung genau einer Pauschalreise absendet und kein Kunde mit sowohl gleichem Vorname und Zunamen und Strasse und PLZ und Ort und Geburtsdatum in der Kundendatenbank vorhanden ist, speichert das System Vorname, Zuname, Strasse, PLZ, Ort und Geburtsdatum in der Kundendatenbank ab und legt genau genau eine Reiseliste für den Kunde in der Kundendatenbank an.

    Eingrenzung des Systems: Durch das Regelwerk wurde erkannt, das implizit der Aufbau einer Kundendatenbank gewünscht ist. Aus Gründen der Komplexitätsreduktion wird nur die Speicherung von Kundendaten behandelt. Auf weitere notwendige Pflegetransaktionen wird nicht eingegangen.

65. Das System darf die Kontodaten des Kunden nicht mitspeichern.
    Kommentar: Aus rechtlichen Gründen.

66. Wenn der Sachbearbeiter die Buchung genau einer Pauschalreise absendet, fügt das System das Reiseziel, den Reiseort, das Hotel, den Preis und die Anzahl der Reisenden der zu buchenden Pauschalreise der Reiseliste des Kunde in der Kundendatenbank hinzu.

67. Wenn das System die Buchungsdaten an den CRS-Server sendet und innerhalb von 10 Sekunden keine Erfolgsmeldung vom CRS-Server erhält, soll das System dem Sachbearbeiter den Hinweis „Buchung nicht erfolgt" ausgeben und ermöglichen die Buchung erneut abzusenden.

68. Wenn das System die Buchungsdaten an den CRS-Server sendet und genau eine Fehlermeldung vom CRS-Server erhält, soll das System dem Sachbearbeiter die Fehlermeldung ausgeben und in der Buchungsmaske verbleiben.

3.6.2 Glossar

Abflugsflughafen:
Ein Ankunftsflughafen ist der Flughafen an dem der mit einer Pauschalreise verbundene Hinflug beginnt.

Ankunftsflughafen:
Ein Ankunftsflughafen ist der Flughafen an dem der mit einer Pauschalreise verbundene Rückflug endet.

buchen:
Buchen ist der Prozess einen rechtsverbindlichen Vertrag über eine Reiseleistung zu schließen.

CRS-Server:
Ein CRS-Server ist eine Datenbank im Internet, in der Reiseveranstalter buchbare Reisen eintragen und über den diese Reisen gesucht und gebucht werden können.

Detailinformationen:
Eine Detailinformation gibt Auskunft über die Eigenschaften eines Hotels.
Beispiel: Nähe zum Strand.
Synonym: Genauere Informationen

Hotel:
Ein Hotel wird jeder Pauschalreise als Übernachtungsmöglichkeit zugeordnet.
Beziehung zu „Zusatzoptionen": Hotels können Zusatzoptionen anbieten.
Beziehung zu „Reiseart": Die Themenzuordnung eines Hotels bestimmt die Reiseart einer Pauschalreise.

Hotelkategorie:
Eine Hotelkategorie gibt die Anzahl der Sterne eines Hotels an.
Beispiel: 5-Sterne

Kind:
Ein Kind ist eine Person bis einschließlich 16 Jahren.
Beziehung zu „Reisender": Ein Kind kann nur Reisender sein.

Kunde:
Ein Kunde ist eine Person, die Interesse hat eine Reise zu buchen und diese nach der Buchung bezahlt.
Synonym: „Reiseanmelder".
Beziehung zu „Reisender": Ein Kunde kann gleichzeitig Reisender sein.

Preis:
Ein Preis ist der Gesamtpreis einer Reise pro erwachsenem Reisenden.

Reise:
Eine Reise ist eine Leistung eines Reisebüros, welche entweder als Einzelleistung (Flug oder Hotelaufenthalt) oder als Gesamtleistung (Pauschalreise mit Flug und Hotelaufenthalt) verkauft wird.
Ergänzung: An einer Reise können maximal sechs Reisende davon maximal drei Kinder teilnehmen.

Reiseart:
Eine Reiseart leitet sich aus der der Themenzuordnung eines Hotels einer Pauschalreise ab.
Beziehung zu „Reisekategorie": Die Reiseart bestimmt nicht zwangsweise die Reisekategorie („Wellness-Urlaub im Golfhotel").
Synonym: Präferenz (des Kunden).
Beispiel: Golfhotel

Reisebuchung:
Eine Reisebuchung ist ein Vorgang, der das Erfassen aller Informationen, die notwendig sind um eine Reise zu buchen und das Buchen der Reise umfasst.

Reisedauer:
Eine Reisedauer ist die Differenz zwischen End- und Starttermin einer Reise in Tagen.

Reisekategorie:
Eine Reisekategorie ist die thematische Einordnung einer Reise.
Beispiel: Wellness-Reise, Badeurlaub

Reisender:
Ein Reisender ist eine Person, die an einer Reise teilnimmt.

Reiseort:
Ein Reiseort ist die Stadt oder der Ort an dem sich das Hotel einer Pauschalreise befindet.
Beispiel: Florenz

Reisezeitraum:
Ein Reisezeitraum ist der Zeitraum in dem eine Reise stattfindet.

Reiseziel:
Ein Reiseziel ist ein Gebiet innerhalb eines Reiselands.
Beispiel: Toskana

unverbindlich:
Ein „unverbindlicher Wunsch" ist ein Wunsch eines Kunden an ein Hotel, aus dem sich keine rechtsverbindlichen Ansprüche ergeben.
Beispiel: Wünsche wie Zimmer mit Meerblick.

Verpflegungsart:
Eine Verpflegungsart gibt den Umfang des Speisen- und Getränkeangebots eines Hotels wider.
Beispiel: Vollpension, All inclusive

Zusatzoptionen:
Eine Zusatzoption ist eine kostenpflichtige, verbindliche Zusatzleistung eines Hotels oder eines Reiseveranstalters zu einer Pauschalreise.
Beispiel: Massagen, Auslandskrankenversicherung

3.7 Einsatz und Bewertung der Regeln

Die vorangegangenen Kapitel zeigten die Iterationen des Analyseprozesses und dessen Ergebnis. Dabei wurde verfolgt, wie sich Anforderungen durch das Regelwerk veränderten. Nachfolgend wird die Perspektive gewechselt und erläutert, mit welchen Regeln, welche Fehler in den Anforderungen entdeckt werden konnten.

3.7.1 Einzelbewertung der Regeln

Gleichzeitig wird die Anwendbarkeit jeder Regel und deren Nutzen bewertet. In den Beispielen wird - wo es möglich ist - versucht, nur die aktuell betrachtete Regel anzuwenden. Es entstehen damit Zwischenformen von Anforderungen, die in den vorhergehenden Ausführungen nicht zu finden sind, da dort meist mehrere Regeln angewendet wurden. Wird in Beispielen auf Regeln mit einer Nummer verwiesen, bezieht sich diese Nummer auf die ursprünglichen Anforderungen aus Kap. 3.5.

Oftmals wird auf die Häufigkeit der Regelanwendung eingegangen. Dies bezieht sich auf die entdeckten Fehler in der Analyse der ursprünglichen Anforderungen bzw. der ersten Umformulierung der Anforderungen. Während der zweiten Umformulierung, in der die Antworten des Anforderungsstellers in die Regeln eingearbeitet wurden, wurde darauf geachtet das *alle* Regeln befolgt werden. Eine Auflistung der dort angewendeten Regeln ergibt damit keinen Sinn.

Um den Bezug zu den Regeln zu erleichtern, werden zunächst jeweils der Regelsatz und die Regelanweisung angegeben.

1: Formulieren Sie jede Anforderung im Aktiv
Die Aktivformulierung hat den Vorteil, dass der Täter, also die ausführende Person oder Einheit, in der Anforderung angegeben werden muss. Dies ist gerade bei Anforderungen entscheidend, da hier wichtig ist, ob die Aktivität vom System, vom Nachbarsystem oder vom Benutzer durchgeführt wird. Auf diese Weise erhält die Anforderung einen höheren Informationsgehalt und das Prozesswort wird näher spezifiziert, da angegeben wird, wer den Prozess durchführt.

Regel 1 wurde sehr häufig eingesetzt, da sich lediglich vier Anforderungen (Nr. 8, 9, 10, 22) im Aktiv befanden. Als Beispiel für die Anwendung können daher nahezu alle Anforderungen dienen. So wurde Anforderung Nr. 7

> *„Zusätzlich kann noch eine Hotelkategorie, der Preis und die Verpflegungsart ausgesucht werden“*

durch Anwendung der Regel zu

> *„Das System ermöglicht es die Hotelkategorie, den Preis und die Verpflegungsart anzugeben.“*

Wie zu erkennen ist, enthält die Anforderung damit die Information über das Subjekt des Satzes. Durch die Umformung wird ferner das Fehlen von weiteren Informationen offensichtlich. Das Projektteam erachtet Regel 1 daher als sehr wichtig. Es geht auch davon aus, dass sie in der Praxis ähnlich häufig eingesetzt wird.

Die Interpretation des Regelsatzes ist nicht eindeutig, da keine Angabe über die bevorzugte Möglichkeit einer Aktivformulierung gegeben wird. So kann eine Anforderung aus Benutzersicht („Der Sachbearbeiter wählt über das System die Hotelkategorie …“) oder aus Systemsicht („Das System ermöglicht dem Sachbearbeiter …“) formuliert werden. In der Simulation wurde analog zu den Beispielen in (Rupp07) die zweite Variante gewählt. Diese wirkt zwar umständlicher, entbindet den Leser jedoch davon, die nötigen Funktionalitäten eines Systems abzuleiten. Eine präzisere Formulierung der Regel wäre wünschenswert.

Bei der Regelanwendung kann es Probleme bereiten, dass das Subjekt zunächst durch eine Annahme ergänzt werden muss. Durch die Regel werden Anforderungen zudem teilweise unnatürlich und eintönig. Sie sind jedoch weiterhin ausreichend gut verständlich. Ansonsten ist die Regel verständlich, außerdem ist leicht zu erkennen, wann sie angewendet werden soll.

2: Drücken Sie Prozesse durch Vollverben aus
Jeder Prozess sollte durch ein Vollverb ausgedrückt werden. Adjektive oder aufwändige Phrasen verschleiern nur den Prozess und lassen die eigentlich durch die Anforderung geforderte Funktionalität in den Hintergrund treten. Vollverben verlagern zudem weitere Satzbestandteile, damit sie näher spezifiziert sind.

Regel 2 wurde nur in zwei Anforderungen (Nr. 8, 13) angewendet. Im Fall von Anforderung Nr. 13 wurde dann sogar auf eine Umformulierung der Anforderung verzichtet, da das verwendete Substantiv „Reisebuchung" dort für einen Vorgang steht, der an anderer Stelle ausführlich spezifiziert wird und damit zur Komplexitätsreduktion beiträgt. Es wird davon ausgegangen, dass die Regel in komplexeren und längeren Anforderungsdokumenten etwas häufiger zum Einsatz kommt. Durch die Anwendung wurde aus Anforderung Nr. 8

„Nach der Selektierung nach den Daten des Kunden …"

schließlich

„Nachdem der Sachbearbeiter die Daten des Kunden eingegeben hat"

Es ist zu erkennen, dass damit der geschilderte Vorgang deutlicher wird. Außerdem wird offensichtlicher, dass weitere Informationen in der Anforderung fehlen (Subjekt, Art und Ort der Eingabe etc.).

Das Verständnis der Regel bereitet keine Probleme. Ihr Zutreffen ist vor allem dann leicht erkennbar, wenn Prozesse durch Substantive ausgedrückt werden.

Nach Meinung des Projektteams überlappt Regel 2 stark mit Regel 17, da in beiden Fällen Nominalisierungen rückgängig gemacht werden sollen, wobei Regel 2 auch andere Fälle der Verschleierung eines Prozesses abdeckt.

3: Decken Sie unvollständig spezifizierte Prozesswörter auf
Bestimmen Sie die Verben in einer Anforderung und stellen Sie fest, ob ein Satz, der mehrere Akteure enthält, mit diesen Verben vorstellbar wäre. Ist dies der Fall, so ist aus dem Originalsatz Information getilgt worden. Ist die fehlende Information wissenswert, sollten Sie gezielt danach fragen.

Regel 3 wurde bei nahezu jeder Anforderung angewendet, da immer zusätzliche Informationen für die Spezifikation der geforderten Funktionalitäten notwendig erschienen. Die Häufigkeit der Regelanwendung ist auch durch die hohe Anzahl an Passivformulierungen begründet. Nichtsdestotrotz wird ein ähnliches Bild von Anforderungsdokumenten in der Praxis erwartet.
In den Beispielen in (Rupp07) werden aus dieser Regel auch Fragen nach dem „Wie" und sogar nach dem „Wie oft" einer Anforderung abgeleitet, obgleich die Regelanweisung lediglich nach den beteiligten Objekten und Personen untersucht. Das Projektteam entschied, der engeren Auslegung zu folgen und die Frage nach dem „Wie" der Regel 5 zuzuordnen.
Als Anwendungsbeispiel kann Anforderung Nr. 7 dienen, in der aus

> *„Das System ermöglicht eine Hotelkategorie auszuwählen."*

folgende Anforderung wurde:

> *„Das System ermöglicht dem Sachbearbeiter eine Hotelkategorie auszuwählen."*

Auch diese Regel wird als sehr wichtig erachtet. Die Notwendigkeit ihrer Anwendung ist im Normalfall leicht erkenntlich.
Die Definition des Begriffs „Prozessworte" und die Klärung der Abhängigkeit zu Regel 5 in den Beispielen in (Rupp07) würde das Verständnis jedoch erleichtern.

4: Ermitteln Sie unvollständige Vergleiche und Steigerungen
Bestimmen Sie die Vergleiche und Steigerungen in einer Anforderung. Enthalten sie den Bezugspunkt, auf den sich der Vergleich oder die Steigerung bezieht? Ist die Abweichung überhaupt messbar? Wenn ja, mit welchen Mitteln (Messmethode)? Mit welcher Genauigkeit kann gemessen werden?

Regel 4 wurde im Anforderungstext lediglich einmal angewendet (Anforderung Nr. 10). Es wird davon ausgegangen, dass die Häufigkeit auch in längeren Texten nicht zunimmt.

In Anforderung Nr. 10 wurde eine Detailanzeige gefordert, die „genauere Informationen" liefert. Der Bezugspunkt zu „genauer", nämlich die Informationen in der Ergebnisliste, wurde dort jedoch nicht angegeben.

Eigentlich wollte der Anforderungssteller durch die Regel die Anzeige von Texten fordern, die vom CRS-Server abgerufen werden. Durch das Fordern der Qualität einer Funktionalität („genauer") konnte so eine implizite Anforderung erkannt werden. Eine Umformulierung der Anforderung durch Hinzufügen des Bezugspunktes unterblieb, statt dessen wurde der Begriff „Detailinformationen" eingeführt und im Glossar definiert.

Das Projektteam ist der Meinung, dass mit der Regel bestimmte implizite Anforderungen gut zu entdecken sind. Ferner kann die Regel auf die Vermischung von funktionalen und nicht-funktionalen Anforderungen hindeuten, bspw. wenn Bedienqualitäten in Anforderungen vorkommen.

Es ist gut zu erkennen, wann die Regel eingesetzt werden muss, auch ist sie leicht zu verstehen.

5: Klären Sie Mögliches und Unmögliches
Stellen Sie bei Aussagen, die eine Möglichkeit oder auch eine Unmöglichkeit beschreiben (siehe Signalwörter), folgende Frage: Was macht das genannte Verhalten möglich bzw. unmöglich? Signalwörter: „kann", „darf", „es ist möglich", „es ist außerstande"

Regel 5 wurde in jeder Anforderung angewendet (oftmals zusammen mit Regel 3). Als Beispiel für die Anwendung mag Anforderung Nr. 4 dienen, in der aus

„Als Zusatzoptionen kann … der Reiseort angegeben werden"

die Anforderung

„Als Zusatzoptionen kann über eine Auswahlliste … der Reiseort festgelegt werden"

wurde. Das Verständnis der Regel fiel dem Projektteam schwer. Ziel ist es nach (Rupp07), für Möglichkeiten, die in Anforderungen ausgedrückt werden, die dazu nötigen Abläufe anzugeben. Dazu soll nach Verben gesucht werden, die eine Möglichkeit ausdrücken („Modaloperatoren der Möglichkeit"). Als Signalwörter werden „kann", „darf", „es ist möglich" angegeben.
Falls der Begriff „Möglichkeit" weit ausgelegt wird, drückt nahezu jede Anforderung eine Möglichkeit aus, nämlich eine Funktion, Eingabemöglichkeit o. ä. Diese weite Interpretation scheint sinnvoll, da für jede Anforderung geklärt werden sollte, wie diese umgesetzt werden kann. Das Vorhandensein bestimmter Modaloperatoren ist demnach nicht ausschlaggebend, da Anforderungen wie nachfolgt unterschiedlich formuliert werden können und dennoch den gleichen Kerninhalt besitzen.

Anwendung der Regel (in (Rupp07) aufgeführtes Signalwort):

◇ Als Zusatzoptionen kann ausgewählt …

Anwendung der Regel (in (Rupp07) nicht aufgeführtes Signalwort):

◇ Das System ermöglicht dem Sachbearbeiter als Zusatzoptionen …

Keine Anwendung der Regel (kein Signalwort):

◇ Das System soll dem Sachbearbeiter die Zusatzoptionen anbieten …

◇ Der Sachbearbeiter wählt …

6: Prüfen Sie Modaloperatoren der Notwendigkeit auf benötigte Ergänzungen
Überlegen Sie sich zu jeder durch einen Modaloperator der Notwendigkeit (siehe Signalwörter) spezifizierten Aussage, ob Sie zusätzlich ein Verhalten für den Ausnahmefall spezifizieren müssen. Signalwörter: „müssen", „sollen", „sollte", „es ist notwendig"

Regel 6 wurde siebenmal angewendet, wobei lediglich bei der Hälfte eines der Signalwörter vorhanden war. In Anforderung Nr. 16

> *„Danach müssen die Kundendaten erfasst werden"*

wurde bspw. geklärt, dass bei Nichtangabe von Kundendaten dennoch ein Absenden der Buchung möglich ist, da der CRS-Server die Kundendaten auf Vollständigkeit bzw. Vorhandensein prüft und ggf. eine Fehlermeldung zurückgibt.

Wie auch in Regel 5 setzt die Anwendung der Regel das Vorhandensein von Modaloperatoren - hier der Notwendigkeit - voraus. Als Signalwörter werden „müssen", „sollen", „sollten" und „es ist notwendig" angegeben. Sind diese vorhanden, soll ein Verhalten im Ausnahme- bzw. Fehlerfall spezifiziert werden.

Analog zu den vorhergehenden Ausführungen ist das Projektteam auch hier der Meinung, dass durch die Signalwörter eine zu starke Einschränkung vorgenommen wird. Zwar ist es sehr wahrscheinlich, dass bei Vorhandensein der Signalwörter eine Fehlerbehandlung spezifiziert werden muss, jedoch ist dies auch ohne das Vorhandensein oftmals nötig. Ein typischer Fall ist die Übertragung von Daten wie in Anforderung Nr. 24

> *„Danach kann die Buchung abgeschickt werden und geht dann zum Reiseveranstalter über."*

Hier ist kein Modaloperator der Notwendigkeit vorhanden, dennoch sollte die Regel eingesetzt werden.

Das Verständnis der Regel ist gut. Mit den vorgegeben Modaloperatoren ist jedoch nicht immer ersichtlich, wann sie eingesetzt werden sollte.

7: Finden Sie implizite Annahmen
Bestimmen Sie das Hauptverb eines Satzes und bilden Sie eine neue Oberflächenstruktur durch Negation dieses Verbs. Danach fragen Sie sich, welche Aussagen wahr sein müssen, damit beide Oberflächenstrukturen einen Sinn ergeben. Alle Aussagen, die Sie dabei entdecken werden, sind unter Umständen nicht formulierte Annahmen, die geklärt und expliziert werden müssen.

Regel 7 wurde zehnmal angewendet und deckte jeweils implizite Anforderungen auf. So konnte bspw. in Anforderung Nr. 10

> *„Das System stellt dem Kunden genau eine Detailanzeige zu jedem Hotel bereit."*

geklärt werden, dass eine Liste mit Detailanzeigen vorhanden sein muss, um diese Anforderung erfüllen zu können. Diese befindet sich auf dem CRS-Server und soll abgerufen werden.

Es wird davon ausgegangen, dass in längeren Texten die Häufigkeit abnimmt, mit der durch die Regel implizite Anforderungen entdeckt werden können, da diese dort an anderen Stellen spezifiziert werden. Die Anwendbarkeit dieser Regel dürfte dabei auch stark nachlassen, da genau dies dann für jede mögliche implizite Annahmen überprüft werden muss.

Trotz der Wichtigkeit, implizite Anforderungen zu erkennen und trotz der vielen neuen Anforderungen, die mit der Regel gefunden wurden, erachtet das Projektteam die Regelanweisung als ungeeignet. Das Projektteam kann einerseits nicht nachvollziehen, warum eine Negation des Hauptverbs notwendig sein soll, da auch ohne die Negation die Aussagen ermittelt werden können, die nötig sind, damit die Anforderungen Sinn ergeben. Andererseits ergeben sich durch die Regel eine Vielzahl von Aussagen, die korrekt sein müssen. Es fehlt aber eine Unterstützung um zu erkennen, welche in neue Anforderungen resultieren sollen.

Wann und wie die Regel eingesetzt wird, ist damit nicht offensichtlich.

8: Überprüfen Sie das Verhältnis der Objekte
Überprüfen Sie, ob das Verhältnis zwischen mehreren Objekten etwas Wesentliches darstellt. Falls dies der Fall ist, formulieren Sie für das Verhältnis eine eigene Anforderung. Falls dies nicht der Fall ist, formulieren Sie für das Verhältnis einen einzigen Begriff (geschlossene Einheit). Der verwendete Begriff ist dann allerdings meist eine Nominalisierung, zu der eine grobe Definition hinterlegt sein muss.

Regel 8 wurde viermal angewendet und konnte dabei ebenfalls implizite Anforderungen aufdecken. So wurde bspw. in Anforderung Nr. 12

> *„Bei der Reisebuchung selbst können noch Zusatzoptionen hinzugefügt werden."*

durch die Prüfung des Verhältnisses von Reisebuchung zu Zusatzoptionen aufgedeckt, dass unterschiedliche Optionen pro Reise zur Verfügung stehen, die vom CRS-Server abgerufen werden müssen.

Ähnlich wie bei Regel 7 wird davon ausgegangen, dass hier eine Abnahme der Anwendungshäufigkeit in längeren Texten gegeben ist und dass die Anwendung der Regel dort auch schwieriger sein dürfte.

Im Gegensatz zu Regel 7 ist die Verständlichkeit von Regel 8 jedoch gut. Nach Meinung des Projektteams ist es dennoch oftmals schwer zu erkennen, wann die Regel eingesetzt werden muss.

9: Bestimmen Sie Universalquantoren
Bestimmen Sie die Universalquantoren einer Anforderung und fragen Sie jeweils, ob das geforderte Verhalten des Systems für wirklich alle Objekte aus der Menge gelten soll, die durch den Quantor zusammengefasst werden. Vielleicht gibt es Ausnahmen, die Sie zusätzlich spezifizieren müssen. Untersuchen Sie jeden Satz dahingehend, ob eine Menge an Objekten ausdrücklich definiert ist, für die das spezifizierte Verhalten gelten soll. Signalwörter: „nie", „immer", „kein", „jeder", „alle", „irgendeiner", „nichts"

Der Anforderungstext enthielt keine Universalquantoren, womit der erste Teil der Regel nicht angewendet werden konnte. Es ist davon auszugehen, dass dies auch typisch für reale Anforderungsdokumente ist. Das Vorhandensein von Universalquantoren, deren Klärung dann auf eine Generalisierung hinweist, würde bedeuten, dass ein Anforderungssteller „bewusst" verallgemeinert. Viel wahrscheinlicher ist hingegen, dass Spezialfälle einfach getilgt werden, wodurch auch Universalquantoren fehlen. Ob der erste Teil der Regel, der auf Universalquantoren prüft, also überhaupt notwendig ist bleibt anzuzweifeln.

Vielmehr sollten für jede Anforderung die betroffenen Objekte und die Häufigkeit der geforderten Funktionalität bestimmt werden.

Die betroffenen Personen - jeder Sachbearbeiter im Reisebüro - wurden stellvertretend für das gesamte Anforderungsdokument in Anforderung Nr. 1 bestimmt und in die Anforderungen eingearbeitet. In komplexeren Systemen ist hier mit Sicherheit von einer größeren Differenzierung der Funktionalitäten und damit von einer häufigeren Regelanwendung auszugehen.

Auf die Angabe der Häufigkeit wurde verzichtet. Diese hätte bedingt, dass in einem Großteil der Anforderungen erwähnt wird, dass die jeweilige Funktionalität „immer" zur Verfügung stehen soll. Es sollte ausreichen, dies in der Anforderungsanalyse zu überprüfen und nur Abweichungen von einer dauerhaften Verfügbarkeit ausdrücklich hervorzuheben.

Nach Meinung des Projektteams ist die Regel unklar formuliert, da der erste Satz der Regel nicht notwendig ist, jedoch dominant wirkt. Durch ihn wird auch die Anwendbarkeit unnötig erschwert.

Das Überprüfen der betroffenen Objekte bzw. Personen einer Anforderung erscheint wichtig. Inwieweit aber jede Erkenntnis der Prüfung (z. B. „immer für jeden Benutzer") explizit in jeder Anforderung ausgedrückt werden muss, bleibt fraglich.

10: Verwenden Sie nur definierte quantitative Angaben
Verwenden Sie als quantitative Angaben zum Beispiel nur "alle", "jeder/jedem", "entweder", "immer", "oder", und "kein" im Deutschen.

Regel 10 untersucht Anforderungen nicht nach sprachlichen Effekten, sondern gibt vor, auf welche Weise quantitative Angaben formuliert werden sollen.

Da bei der Umformulierung, wie erwähnt, auf das Einhalten aller Regeln geachtet wurde, erschien es nicht sinnvoll den Einsatz gesondert zu dokumentieren, weswegen der Einsatz von Regel 10 in der Simulation nicht auftaucht.

Dennoch wurde die Regel nahezu in jeder Anforderung eingesetzt. Das Projektteam erachtet sie auch als notwendig, da durch sie Verständlichkeit und Eindeutigkeit der Anforderungen unterstützt werden.

Das Verständnis der Regel leidet jedoch darunter, dass in Regel 14, 15 und 16 ähnliche oder widersprüchliche Aussagen getroffen werden. Das Projektteam findet, dass die vier angegebenen Regeln derart stark zusammenhängen, dass sie in einer Regel zusammengefasst werden sollten. Davon würde sowohl die Erlernbarkeit, wie auch die Anwendbarkeit des Regelwerks profitieren.

11: Ermitteln Sie unvollständig spezifizierte Bedingungen
Bestimmen Sie die Bedingung(en) in der Anforderung und überprüfen Sie, ob sowohl für den Fall, dass die Bedingung eintritt, ein Verhalten spezifiziert ist (dann-Zweig), als auch dafür, dass sie nicht eintritt (sonst-Zweig). Stellen Sie sich zudem die zusätzlichen Fragen: "Sind alle möglichen Entscheidungskriterien (Bedingungen) aufgezählt?" und "Sind alle möglichen Varianten geschildert?" Signalwörter: „wenn", „dann", „falls", „im Falle von", „abhängig von"

Regel 11 wurde an fünf Stellen angewendet. Dort wurde die Regel jeweils sehr frei ausgelegt, da Bedingungen nur implizit in den Anforderungen vorhanden waren. Dementsprechend trafen auch die angegebenen Signalwörter nicht zu. Bei der Anwendung wurde bspw. in Anforderung Nr. 8

> *„Nach der Selektierung nach den Daten des Kunden gibt das System eine Ergebnisliste aus mit den vorhandenen Hotels mit genau den Kriterien."*

die Notwendigkeit erkannt, zu hinterfragen, was im Falle einer leeren Ergebnisliste passieren sollte. Eine ähnliche Anwendung geschah in den anderen Fällen, in denen die Regel angewendet wurde. Das Projektteam geht davon aus, dass die Signalwörter und die Anwendung der Regeln in realen Anforderungsdokumenten häufiger auftauchen. Es wird jedoch wie in Regel 9 angenommen, dass Generalisierung und Tilgung häufig gemeinsam auftreten und damit nur der Standardfall spezifiziert wird, wobei auf Eintrittsbedingungen gar nicht, anstatt nur nicht vollständig eingegangen wird.

Die Bedeutung und Notwendigkeit der Regel ist nach Meinung des Projektteams gegeben. Auch ist sie verständlich. Schwierig stellt es sich jedoch dar, zu erkennen, wann die Regel eingesetzt werden muss.

12: Hinterfragen Sie Substantive ohne Bezugsindex
Fragen Sie für jedes Substantiv der Anforderung, ob es eigentlich eine spezifische Person, eine spezifische Personengruppe, einen spezifischen Gegenstand oder eine Gruppe von Gegenständen der Welt bezeichnen sollte. Sie können bei Substantiven, die eine nicht genau einzugrenzende Menge von Objekten beschreiben, die folgenden Fragen stellen: "Wer ... genau?", "Was ... genau?", "Welcher Teil der genannten Menge?" Erweitern Sie solche Substantive dann durch eine genau festlegende Ergänzung.

Regel 12 wurde in nahezu jeder Anforderung eingesetzt, da bei vielen verwendeten Substantiven nicht klar war, worauf sie sich beziehen bzw. wie das dahinter stehende Objekt bestimmt ist.

Durch die Regelanwendung wurden in Anforderung Nr. 2

> *„Zuerst wird der Reisezeitraum festgelegt, danach wird die Anzahl der Personen festgelegt und ob Kinder teilnehmen.“*

die Substantive „Reisezeitraum‟, „Personen‟ und „Kinder‟ genauer bestimmt, so dass die Anforderung

> *„Zuerst wird der Reisezeitraum der gesuchten Reise festgelegt, danach wird die Anzahl der an der gesuchten Reise teilnehmenden Personen festgelegt und ob Kinder an der gesuchten Reise teilnehmen.“*

entstand. Diese ist auch ohne Kontextwissen verständlich.

Ein anderes Beispiel stellt Anforderung Nr. 16 dar,

> *„Danach müssen die Kundendaten erfasst werden.“*,

in der geklärt wurde, dass Kundendaten der Vor- und Zuname, Geburtsdatum, sowie Adresse sind.

Das Projektteam erachtet die Regel als sehr sinnvoll, da durch sie bis zu einem gewissen Grad kontextunabhängige Anforderungen entstehen und Klarheit über verwendete Substantive geschaffen wird.

Die Notwendigkeit ihrer Anwendung ist im Normalfall leicht erkennbar.

Die Regel ist zunächst jedoch schwer verständlich, da die Bezeichnung „Bezugsindex‟ in (Rupp07) nicht erklärt wird und die angegeben Beispiele für kein besseres Verständnis sorgen.

13: Verwenden Sie Substantive stets in der Einzahl
Die in der Anforderung vorkommenden Substantive sollten zudem in der Einzahl verwendet werden, außer der geforderte Sachverhalt bezieht sich ausschließlich auf eine Gruppe in ihrer Gesamtheit.

Regel 13 wurde dreimal angewendet. So stand bspw. in Anforderung Nr. 20

> *„Dabei muss dann der Name, das Geschlecht und bei Kindern das Alter angegeben werden."*

das Wort „Kindern" im Plural und wurde daher durch die Formulierung „bei jedem Kind" ersetzt. Dabei kam auch Regel 14 zum Einsatz. An vielen anderen Stellen standen weitere Substantive im Plural, bei denen die Regel jedoch nicht griff, da die Gesamtzahl der jeweiligen Objekte gemeint war. So wurde bspw. in Anforderung Nr. 15 „unverbindliche Wünsche des Kunden" nicht abgeändert.

Trotz des eher geringen Einsatzes erachtet das Projektteam diese Regel als enorm wichtig, da dadurch Mehrdeutigkeiten, die beim Einsatz des Plurals entstehen, gut vermindert werden können.

Bei der Anwendung kann es jedoch Probleme bereiten zu entscheiden, wann ein Substantiv im Plural verbleiben soll, da es sich auf die Gesamtheit einer Gruppe bezieht. Die Beispiele in (Rupp07) sind hier irreführend bzw. nicht korrekt. Dort wurde bspw. die Anforderung „Allen Gästen, die keinen Menüplan gewählt haben, sollen sämtliche … Speisen des Restaurants auf dem Mobilteil in der Liste angezeigt werden." als korrekt bezeichnet. Diese Anforderung ist jedoch mehrdeutig und kann so interpretiert werden, dass sich jene Gäste ein Mobilteil teilen auf dem sie gemeinsam eine Liste angezeigt bekommen. Eindeutiger wäre hier die Formulierung „Jedem Gast, …, sollen sämtliche Speisen auf dem Mobilteil angezeigt werden". Es wäre wünschenswert, wenn diese Problematik in (Rupp07) genauer aufgegriffen werden würde.

14: Machen Sie konkrete Angaben über die Menge an Objekten
Falls es sinnvoll ist, sollten Sie den Artikel je nach Bedeutung durch "jeder", "alle" oder "genau ein" ersetzen. Angaben über die Menge der Objekte, die Substantive umschreiben, werden nicht nur durch Mehr- oder Einzahl ausgedrückt, sondern durch spezifische Angaben, die den Artikel (der, die, das, ein, eine, ein, ...) vor dem Substantiv ersetzen, noch näher spezifiziert.

15: Verwenden Sie unbestimmte Artikel nur in Definitionen
Verwenden Sie „ein" (unbestimmter Artikel) nur vor einem Substantiv, das damit gerade definiert wird.

16: Verwenden Sie den bestimmten Artikel vor einem Substantiv, das schon definiert ist
Ist keine Mengenangabe vor einem Substantiv notwendig oder sinnvoll, sollte in einer Anforderung immer der bestimmte Artikel (der, die, das oder eine deklinierte Form wie dem, den, das) verwendet werden

Regeln 14 bis 16 sind nach Meinung des Projektteams derart stark voneinander abhängig, dass eine gesonderte Betrachtung nicht sinnvoll erscheint. Regel 14 wurde in jeder Anforderung verwendet, da immer eine zusätzliche Klärung der Mengen der beteiligten Objekte als notwendig erschien. Dementsprechend wurde Regel 15 in jeder Definition eines Substantivs angewendet.

Das Projektteam geht davon aus, dass auch in realen Anforderungstexten eine derart häufige Verwendung gegeben ist, da die explizite Mengenangabe von Substantiven nicht dem normalen Sprachgebrauch entspricht und damit nachträglich ergänzt werden muss. Als Beispiel für die Anwendung kann Anforderung Nr. 7 dienen

> *„Zusätzlich kann noch eine Hotelkategorie, ... ausgesucht werden."*

in der die Menge an Hotelkategorien bestimmt wurde und daraus die Anforderung

> *„Das System ermöglicht jedem Sachbearbeiter über je genau eine Auswahlliste in der Suchmaske optional genau eine Hotelkategorie und optional genau eine Verpflegungsart für alle zu suchenden Pauschalreisen auszuwählen."*

entstand. Die Regeln 14 bis 16 werden als äußerst sinnvoll erachtet und tragen nach Meinung des Projektteams stark zur Eindeutigkeit von Anforderungen bei. Die Anwendung gestaltet sich teilweise schwierig, da erkannt und entschieden werden muss, wo eine Mengenangabe und wo ein bestimmter Artikel sinnvoll ist. Auch ist das Umformulieren von Anforderungen

teilweise schwierig. Die entstehenden Anforderungen sind jedoch weiterhin gut verständlich und tragen mehr Informationen, obgleich sie etwas unnatürlicher wirken.

Das Verständnis der Regeln leidet wie erwähnt unter der Aufteilung auf drei Regeln. Wie bereits erwähnt ist auch eine starke Verbindung zu Regel 10 gegeben, die für zusätzliche Unklarheiten sorgt.

17: Hinterfragen Sie Nominalisierungen
Überprüfen Sie jedes Substantiv im Satz und fragen Sie sich, ob nicht ein Verb möglich gewesen wäre, das einen Vorgang beschreibt und das ähnlich klingt/aussieht und dem Substantiv in der Bedeutung ähnelt. Dazu zwei Tests: 1. Passt das Substantiv sinnvoll in die Phrase "ein(e) andauernde(r) ..." (im Sinne von kontinuierlich), so handelt es sich um einen Vorgang, der normalisiert wurde. 2. Beschreibt ein Substantiv etwas, dass man nicht anfassen kann? Auch hier handelt es sich wahrscheinlich um einen nominalisierten Prozess. Kann eine der beiden Fragen mit "Ja" beantwortet werden, so müssen Sie das Substantiv dahingehend hinterfragen, ob nicht mit dem unterschlagenen Vorgang auch Informationen verloren gegangen sind.

Regel 17 wurde zweimal eingesetzt. Wie bereits in Regel 2 erwähnt, geht das Projektteam davon aus, dass die Regel in längeren und komplexeren Texten häufiger auftaucht, aber auch dass diese Regel eigentlich mit Regel 2 bereits abgedeckt ist. Durch Regel 17 wurde bspw. in Anforderung Nr. 13

> *„Bei der Reisebuchung selbst können noch Zusatzoptionen hinzugefügt werden.“*

erkannt, dass der Buchungsvorgang einer Reise als kurzes Ereignis dargestellt wird. Eine Umformulierung geschah jedoch nicht, da dieser Vorgang in weiteren Anforderungen genauer spezifiziert wird und nach (Rupp07) dies dann nicht nötig ist. Die Verständlichkeit der Regel ist gegeben.

Die Regelanweisung ist jedoch nicht brauchbar. Während mit der ersten Prüfung, Prozesse gut identifiziert werden können, trifft die zweite Prüfung auf zu viele Substantive zu, die keinen Prozess darstellen (z. B. Information).

18: Ersetzen Sie die Funktionsverbgefüge durch einfache, direkte Vollverben
Funktionsverbgefüge führen zu einem mit Substantiven überladenen, schwer verständlichen Stil und bewirken oft, dass ein Vorgang in passivischer Sichtweise beschrieben wird. Viele Sätze sind unnötig indirekt ausgedrückt, sodass die eigentliche Tätigkeit, die normalerweise durch ein Vollverb beschrieben ist, in den Hintergrund gerät.

Regel 18 wurde im Anforderungstext nicht angewendet. Es wird davon ausgegangen, dass dies aus der mündlichen Erhebungsform folgt.

Auch wenn im gegebenen Fall keine Beispiele für Funktionsverbgefüge vorhanden sind, geht das Projektteam davon aus, dass die Regel durchaus Relevanz und Notwendigkeit besitzt. Durch den Einsatz werden Vorgänge klarer. Die Verständlichkeit der Regel ist gegeben, ebenso ist offensichtlich, wann die Regel zutrifft.

19: Vermeiden Sie Redundanz
Vermeiden Sie es, etwas doppelt auszudrücken. Entfernen Sie Teile des Satzes, die Sie ohne Bedeutungsverlust straffen können.

20: Vermeiden Sie Floskeln
Streichen Sie floskelhafte Wörter und Wendungen oder drücken Sie diese kürzer aus.

Regel 19 wurde nicht und Regel 20 wurde fünfmal eingesetzt. Nach Meinung des Projektteams hätten die fünf Verwendungen auch der Regel 19 zugeschrieben werden können, da der Inhalt der Regeln nahezu identisch ist.

Es ist davon auszugehen, dass die Häufigkeit der Anwendung stark vom Sprachgebrauch des Anforderungsstellers und von der Erhebungsart der Anforderungen abhängig ist. Im Fall von Anforderung Nr. 4

> *„Als Zusatzoptionen kann noch dazu der Abflugs- und Ankunftsflughafen, die Reiseart, das Reiseziel und der Reiseort angegeben werden."*

wurde *„noch dazu"* aus der Anforderung entfernt.

Der Gewinn an Eindeutigkeit ist dabei eher gering, ähnlich ist es in den anderen Fällen der vorliegenden Anforderungen. In den Beispielen in (Rupp07) in denen Floskeln wie „Vor dem Hintergrund …" entfernt werden, kann jedoch durchaus ein Zugewinn an Verständlichkeit erkannt werden.

Das Projektteam ist der Meinung, dass die Wichtigkeit dieser Regel stark vom vorliegenden Anforderungsdokument abhängt. Im gegebenen Fall ist die Bedeutung der Regel eher gering.

Die Verständlichkeit der Regel ist gegeben. Die Notwendigkeit des Einsatzes bleibt jedoch teilweise umstritten, da es stark vom Sprachgefühl des Anwenders des Regelwerks abhängt, was als Floskel bezeichnet werden kann. Inwieweit das Streichen von zu langen oder überflüssigen Wendungen auf zwei Regeln verteilt werden muss, bleibt fraglich.

21: Überprüfen Sie Nebensätze
Überprüfen Sie Nebensätze, die eine Begründung, Absicht oder Folge enthalten. Falls darin keine wichtigen Informationen verborgen sind, können Sie diese als Kommentar herauslösen, ansonsten müssen Sie daraus eigenständige Anforderungen formulieren.

Regel 21 wurde viermal angewendet. Es ist davon auszugehen, dass in schriftlich erhobenen Anforderungen die Häufigkeit steigt, da hier komplexere Satzstrukturen zu erwarten sind. Bei der Anwendung der Regel wurde bspw. in Anforderung Nr. 1

> *„Um eine Reise zu buchen, muss zunächst erfasst werden, was für eine Reise der Kunde haben will.“*

der Satzteil *„Um eine Reise zu buchen“* entfernt, da er lediglich eine Begründung darstellt. In diesem Fall ist der Zugewinn an Eindeutigkeit eher gering. Gerade wenn komplexere Satzgefüge aufgelöst werden, kann jedoch davon ausgegangen werden, dass die Verständlichkeit und Eindeutigkeit steigt, da durch das Verbieten von Nebensätzen viele Mehrdeutigkeiten auf Satzebene nicht mehr auftreten können.

Ergänzend zu der Regel heißt es in (Rupp07), dass eine Anforderung nur aus einem Hauptsatz bestehen sollte.

Trotz der Vorteile der Regel erachtet sie das Projektteam als weniger bedeutend, da gerade bei komplexeren Anforderungen der Einsatz mehrerer Hauptsätze oder die Verwendung von Nebensätzen schlussendlich verständlicher ist als ein einziger Satz, der mit Informationen überladen ist.

Ansonsten ist das Verständnis der Regel gut, ebenso ist leicht zu erkennen, wann die Regel eingesetzt werden sollte.

22: Stellen Sie zeitliche Abhängigkeiten klar
Nebensätze, die in einem zeitlichen Verhältnis zum Hauptsatz stehen, müssen Sie in einem Wenn/Falls-Satz mit eventuellen Verneinungen umformulieren. Schlüsselwörter zur Identifizierung temporaler Nebensätze (Nebensätze, die in einem zeitlichen Verhältnis stehen) sind zum Beispiel: "bevor", "während", "nachdem", "solange", "bis", "unterdessen", "solange", "bis", "unterdessen".

Regel 22 wurde fünfmal angewendet, wobei auch Anforderungen betrachtet wurden, die keine Nebensätze, jedoch den Hinweis auf zeitliche Abhängigkeiten zu anderen Anforderungen enthielten. Als Beispiel dient Anforderung Nr. 8

> *„Nach der Selektierung nach den Daten des Kunden gibt das System eine Ergebnisliste aus mit den vorhandenen Hotels mit genau den Kriterien."*

Diese wurde zu

> *„Wenn die Daten des Kunden selektiert wurden, gibt das System eine Ergebnisliste aus mit Hotels mit genau den Kriterien."*

umformuliert. Es wird davon ausgegangen, dass die Häufigkeit der Anwendung in Anforderungen aus der Praxis auch gegeben ist.

Das Ziel, zeitliche Abhängigkeiten eindeutig und verständlich darzustellen, wird als sehr wichtig erachtet. Inwieweit die Regel dabei hilft, ist jedoch anzuzweifeln. So bietet die Regel keine Unterstützung für Abhängigkeiten zwischen Anforderungen. Außerdem ist die Eindeutigkeit nur gegeben, wenn zwei Vorgänge zueinander in Beziehung gesetzt werden sollen. Bereits bei einer Abfolge von drei Vorgängen stößt die Regel an ihre Grenzen.

> *„Wenn der Sachbearbeiter die Buchungsmaske öffnet und das System nicht innerhalb von 5 Sekunden vom CRS-Server die Zusatzoptionen einer Pauschalreise abrufen kann, soll das System dem Sachbearbeiter den Hinweis „CRS-Server nicht erreichbar" ausgeben und in der Ergebnisliste verbleiben."*

Eine Aufteilung in zwei Sätze wäre hier jedoch ebenfalls nicht praktikabel.

Das Verständnis der Regel ist gut, ferner ist leicht erkenntlich, wann die Regel eingesetzt werden muss. Das Projektteam ist jedoch der Meinung, dass die Regel für zu wenige zeitliche Abhängigkeiten geeignet ist.

23: Definieren Sie Substantive nach folgendem Schema
Subjekt (=zu definierender Begriff) + Verb + Objekte + Ergänzungen (Phrasen, Nebensätze)

24: Definieren Sie Eigenschaftswörter nach folgendem Schema
Zu definierendes Eigenschaftswort + Hilfssubstantiv + "ist" + Hilfssubstantiv + Ergänzungen

25: Definieren Sie Verben nach folgendem Schema
Verb im Infinitiv + "ist der Prozess" + Ergänzungen

Aus Regeln 23 bis 25 resultieren 20 Definitionen für Substantive und je eine für Adjektive bzw. Verben. Es wird davon ausgegangen, dass die hohe Zahl im Vergleich zur Länge des Anforderungstextes durchaus exemplarisch ist.

Die Bedeutung der Regeln wird als enorm hoch eingeschätzt. Mit ihnen kann ein Teil des fachlichen Hintergrunds der Anforderungen vermittelt werden, wodurch Zusammenhänge klar werden, aber auch das nötige Wissen für Design und Implementierung eines Systems vermittelt wird.

Trotz des kurzen Anforderungstextes, der zudem von nur einer Person erstellt wurde, konnten Synonyme und damit redundante Anforderungen erkannt werden (z. B. Anforderung Nr. 5). Auch ließen sich durch die Präzisierung der Bezeichnungen fehlende Anforderungen entdecken, wie etwa in Anforderung Nr. 2, in der nur die Eingabe des Reisezeitraums gefordert wurde, aber die Reisedauer vergessen wurde.

Ein weiteres Beispiel ist Anforderung Nr. 1, in der es hieß

> *„Es muss erfasst werden, was für eine Reise der Kunde haben will“.*

Erst durch die Definition von „Reise‟ erkannte das Projektteam, dass eigentlich eine Auswahl verschiedener Reiseleistungen gewünscht war.

> *„Es kann zwischen Hotels, Flügen und Pauschalreisen gewählt werden“*

Für das Projektteam war offensichtlich, wann die Regel angewendet werden musste. In einer realen Umgebung dürfte es jedoch schwerer fallen zu entscheiden, welche Bezeichnungen definiert werden sollen und welche nicht.

3.7.2 Zusammenfassung

Die gewonnenen Erkenntnisse aus der Einzelbewertung lassen sich vereinfacht auch in der nachfolgenden Tabelle zusammenfassen.

Regel	Nutzen	Häufigkeit	Verständlichkeit	Erkennbarkeit
1	+	+	+	+
2	+	-	+	+
3	+	+	~	+
4	+	-	+	+
5	+	+	~	~ [a]
6	+	~	+	~ [a]
7	-	+	-	-
8	+	~	+	-
9	~	-	-	~
10	+	+	~ [b]	~
11	+	~	+	-
12	+	+	~	~
13	+	-	~	+
14	+	+	~ [b]	+
15	+	+	~ [b]	+
16	+	+	~ [b]	+
17	-	-	-	-
18	~	-	+	+
19	~	~	+	~
20	~	~	+	~
21	-	~	+	+
22	~	~	+	+
23	+	+	+	~
24	+	-	+	~
25	+	-	+	~

[a]) bei Verwendung der Signalwörter aus (Rupp07)
[b]) durch starke Abhängigkeiten untereinander

Tabelle 1: Übersicht über die Bewertung der Einzelregeln

Wie zu erkennen ist, werden nahezu alle Anforderungen als notwendig (Nutzen) eingestuft. Differenzierter ist das Bild im Bereich der Verständlichkeit und Erkennbarkeit der Anwendung, wo bei einigen Regeln Defizite auftauchen.

4 Bewertung des Vorgehens

Werden Anforderungen ohne ein Rahmenwerk auf deren Qualität untersucht, geschieht dies im Normalfall intuitiv und nicht zielorientiert. Die Folge sind Mehraufwendungen und nicht erkannte Fehler. Um dem vorzubeugen, sollte ein Regelwerk zur Analyse und Qualitätsprüfung von Anforderungen optimalerweise folgende Eigenschaften aufweisen:

Erlernbarkeit und Eindeutigkeit

Das Regelwerk ist leicht erlernbar und führt zu eindeutigen Ergebnissen, wenn unterschiedliche Personen es anwenden. Es besitzt einen geringen Interpretationsspielraum.

Vollständigkeit bei Minimalität

Das Regelwerk umfasst alle Regeln, die notwendig sind um die häufigsten Fehler in Anforderungen zu entdecken und Anforderungen eindeutig, vollständig und widerspruchsfrei zu formulieren. Es enthält keine überflüssigen Prüfungen oder Regeln.

Methodisches Vorgehen

Der Anwender des Regelwerks wird zielgerichtet durch die Analyse der Anforderungen geführt.

Anwendbarkeit

Das Regelwerk ist sinnvoll in einer realen Umgebung anwendbar. Es benötigt einen angemessenen zeitlichen Aufwand, wird von den Beteiligten der Systemanalyse akzeptiert und überlastet sie nicht.

Um zu überprüfen, inwieweit das SOPHIST-REgelwerk diese Eigenschaften erfüllt, wird nachfolgend das Vorgehen in der Anwendung des Regelwerks anhand der eben aufgestellten Kriterien bewertet. Die Bewertung wurde aufgeteilt in die Sicht des Projektteams und in die Sicht des Anforderungsstellers. Für die Kriterien wurden Leitfragen formuliert, die im Folgenden beantwortet werden. Gegen Ende des Kapitels werden die gewonnen Erkenntnisse zusammengefasst und den hier aufgestellten Eigenschaften gegenübergestellt.

4.1 Erlernbarkeit des Regelwerks

Bei der Erlernbarkeit soll der zeitliche Aufwand für das Erlernen des Regelwerks aufgezeigt werden, sowie dessen Verständlichkeit und Verinnerlichung bewertet werden. Es wird außerdem beurteilt, inwieweit das SOPHIST-REgelwerk auf mündliche Anforderungen angewendet werden kann.

4.1.1 Zeitlicher Aufwand für das Erlernen

Wie lange dauert die Vorbereitung zur Erlernung des Regelwerks?

Grund-legendes Verständnis

Der Betrachtung der Vorbereitungszeit liegt zunächst die Annahme zu Grunde, dass das Regelwerk durch die Lektüre von (Rupp07) selbständig erlernt und im späteren Verlauf analog zur Simulation auf schriftlich fixierte Anforderungen angewendet werden soll. Unter diesen Randbedingungen kann von einem Zeitaufwand von lediglich zwei Stunden ausgegangen werden, um ein für die Anwendung ausreichendes Verständnis zu erreichen. Dies wird gerade durch die knappe Darstellung in (Rupp07) ermöglicht.

Als ausreichend kann dabei angesehen werden, die Regelanweisung zu jedem Regelsatz sinngemäß zu kennen. Da es sich als empfehlenswert gezeigt hat, die Anwendung des Regelwerks durch eine Auflistung der Regeln zu unterstützen, ist es nicht nötig sämtliche Regeln auswendig aufzählen zu können. Dadurch wird der Lernaufwand erheblich gemindert.

Vertieftes Verständnis

Es bleibt hierbei anzumerken, dass ein vertieftes Verständnis der Regeln nicht rein durch die Lektüre, sondern erst durch deren Anwendung erreicht werden kann. Dabei treten gewöhnlich weitere Fragen auf, die durch erneutes Nachlesen in (Rupp07) oder andere Recherchen geklärt werden müssen und zusätzlichen Aufwand bedeuten. Hier wirkt sich vor allem die im Anschluss bewertete Darstellung des Regelwerks in (Rupp07) negativ aus. Der Zeitaufwand für die nachträgliche Lektüre kann mit einer weiteren Stunde angesetzt werden.

4.1.2 Verständlichkeit der Darstellung des Regelwerks

Sind die Regeln verständlich erklärt?

Von Seiten der SOPHIST-Group existieren unterschiedliche Darstellungen des Regelwerks in den unterschiedlichen Auflagen des Buches „Require-

ments-Engineering und -Management" von Chris Rupp (z. B. (Rupp07), (Rupp02)). Außerdem ist unter www.sophist.de eine ältere Version des Regelwerks als pdf-Dokument erhältlich (SOPHIST01). Die nachfolgende Betrachtung orientiert sich ausschließlich am aktuellsten Werk (Rupp07), kann aber auch auf die anderen Darstellungen in den Kernaussagen übertragen werden.

Viele Beispiele

In (Rupp07) werden, wie erwähnt, die Regelsätze und -anweisungen einzeln vorgestellt, wobei meist eine ergänzende, längere Begründung der Regel vorgenommen wird. Hier ist zunächst positiv anzumerken, dass das Regelwerk durch eine Vielzahl an Beispielen erläutert wird.

Ungeeignete Beispiele

Leider ist zu erkennen, dass diese Beispiele nicht in jedem Fall aufschlussreich sind. Zur Illustration kann Regel 7 („Finden Sie implizite Annahmen") dienen. Deren Regelanweisung sagt aus, dass implizite Annahmen ermittelt werden können, indem für die eigentliche Anforderung eine neue Oberflächenstruktur durch Negation des Hauptverbs gebildet wird. Es soll dann gefragt werden welche Aussagen wahr sein müssen, damit die negierte und die nicht negierte Anforderung Sinn ergeben.
Chris Rupp veranschaulicht diese Regel anhand eines sehr fragwürdigen Beispiels. Es lautet „Mein Hund fiel gestern von der Leiter" und negiert dementsprechend „Mein Hund fiel gestern nicht von der Leiter". Dabei behauptet sie, dass aus beiden Aussagen geschlossen werden kann, dass der Sprecher einen Hund haben muss. Warum für das Erkennen dieses Sachverhalts die Verneinung des Ausgangssatzes nötig ist, begründet sie nicht weiter.[4] Es bleibt folglich unklar, wie die Regel ihr eigentliches Ziel unterstützen soll. Auch an anderen Stellen liefern die Beispiele teilweise einen geringen Beitrag zum Verständnis. Als Folge lassen sich diese Regeln nur schwer auf eigene Anforderungen übertragen.

Nicht erkennbarer Regelbezug

Ebenfalls ist häufig nicht erkennbar, auf welche Regel sich ein Beispiel bezieht. So werden bspw. in den Anforderungen zu Regel 3 (die Anforderungen auf getilgte Akteure untersucht) Fragen nach dem „wie" und „wann"

---

[4] Vgl. (Rupp07), S. 155.

einer Funktionalität abgeleitet.[5] Diese beziehen sich aber auf die später vorgestellten Regeln 5 bzw. 9. Durch Beispiele, die weder zu vorhergehenden, noch zu nachfolgenden Regeln gehören und durch die allgemein unübersichtliche Strukturierung des Regelwerks in (Rupp07) wird das Verständnis weiter unnötig erschwert.

Ungelöste Beispiele

Als letzter Kritikpunkt bzgl. der Beispiele kann aufgeführt werden, dass teilweise eine ausführliche Lösung fehlt. So werden lediglich die sich ergebenden Fragen gezeigt, nicht aber wie neue, bessere Anforderungen formuliert werden können.

Unklare Signalwörter

Neben den Beispielen und der Struktur der Darstellung erscheinen weitere Punkte verbesserungsbedürftig. So ist die Wahl der Signalwörter nicht in jedem Fall schlüssig. Bspw. heißt es in Regel 5, dass auf Modaloperatoren der Möglichkeit (kann, darf, es ist (nicht) möglich,...) geachtet werden sollte, um zu erkennen, wann die Regel greift. Chris Rupp verwendet zur Verdeutlichung dieser Regel anschließend eine Anforderung, die auf einem Modaloperator der Notwendigkeit und zwar „soll" aufbaut. Dieser ist allerdings ein Signalwort, das auf die Anwendbarkeit von Regel 6 hinweist.[6] Auch in den drei Beispielen zu Regel 11 taucht lediglich einmal ein angegebenes Signalwort auf.[7] Es stellt sich hier allgemein die Frage, inwieweit die Signalwörter richtig gewählt sind bzw. inwieweit sie bei der Analyse von Anforderungen unterstützen können. Unklarheiten entstehen dadurch, dass nicht geklärt wird, ob Signalwörter zwingend notwendig oder optional für die Anwendung einer Regel sind.

Nicht definierte Fachbegriffe

Ein weiterer verbesserungsbedürftiger Aspekt ist die fehlende Definition von Fachbegriffen aus den Bereichen des Neuro-Linguistisches-Programmierens und der Grammatik. Hiervon sind einige im SOPHIST-REgelwerk vorhanden, deren Kenntnis bei einem durchschnittlichen Anwender nicht vorausgesetzt werden kann. Zumindest diese sollten definiert werden, da ihr Verständnis sehr wichtig für die richtige Auslegung der Regeln ist.

---

[5] Vgl. (Rupp07), S. 150
[6] Vgl. (Rupp07), S. 152 f.
[7] Vgl. (Rupp07), S. 159 f.

Meist erfolgt jedoch nur eine Auflistung von Beispielbegriffen, die unter den Fachbegriff fallen (Extension des Fachbegriffs). Dies ist zwar als Ergänzung zu einer Definition wünschenswert, reicht aber für ein präzises Verständnis alleine nicht aus. Ordentliche Definitionen, bei denen die Merkmale eines Fachbegriffs geklärt werden, werden jedoch kaum vorgenommen oder sind vage und teilweise sogar widersprüchlich.
Als Beispiel kann hier der Begriff „Prozesswort" dienen, unter dem in (Rupp07) Verben, Adjektive und Adverbien und in (SOPHIST01) zudem Substantive zusammengefasst werden. An anderer Stelle wird er in (Rupp07) nur für Verben und in (SOPHIST01) für Verbalphrasen verwendet.[8] Eine ähnliche Verwirrung entsteht bei den Begriffen „Modaloperator der Möglichkeit bzw. Notwendigkeit", die ebenfalls nicht eindeutig bestimmt werden. Bei ihnen tritt zudem die oben erwähnte Unklarheit im Bezug auf Signalwörter auf. Da Modaloperatoren nur durch diese Signalwörter „definiert" werden, ist deren Verständnis ohne weitere Recherche nicht möglich. Weitere Beispiele sind die Begriffe „Universalquantoren" (die fehlerhaft als „Angaben über Häufigkeit"[9] bezeichnet werden) und „Bezugsindex".

Unvollständige Regelanweisungen

Der letzte Kritikpunkt an der Darstellung des Regelwerks ist, dass wichtige Inhalte in Regelanweisungen fehlen oder nur indirekt ausgedrückt werden. So wurde bereits in der Bewertung der Regeln angemerkt, dass in Regel 1 („Formulieren Sie jede Anforderungen im Aktiv") nicht erwähnt wird, dass Anforderungen aus Systemsicht formuliert werden sollen. In Regel 9 („Bestimmen Sie Universalquantoren") tritt in den Hintergrund, dass auch das Fehlen von Universalquantoren überprüft werden soll. In Regel 21 („Überprüfen Sie Nebensätze") fehlt, dass Anforderungen aus einem Hauptsatz bestehen sollen und in Regel 23 („Definieren Sie Substantive …") wird bspw. keine Aussage über die Behandlung oder Suche nach Synonymen getroffen. Die fehlenden Elemente tauchen nur in den Erläuterungen zu den Regeln auf, stellen jedoch nach Meinung des Projektteams zentrale Punkte dar. Das Ergebnis dieses Umstandes ist, dass Anwender des Regelwerks nicht nur die 25 Regeln kennen müssen, sondern den gesamten erläutern-

---

[8] Vgl. (Rupp07) S. 148 ff., (SOPHIST01) S. 11
[9] (Rupp07), S. 157

den Text in (Rupp07). Damit wird sowohl die Anwendbarkeit des Regelwerks erschwert, wie auch die Fehlerwahrscheinlichkeit erhöht.

Ist eine gesonderte, eigene Darstellung der Regeln im Analyseprozess notwendig?

Darstellung für Schulungszwecke

Aus den vielen Kritikpunkten an der Darstellung des Regelwerks in (Rupp07), die sich auf die weiteren Darstellungen übertragen lassen, ergibt sich die Frage, ob eine eigene Darstellung des Regelwerks sinnvoll erscheint. Diese sollte dann die erwähnten Unklarheiten und Interpretationsspielräume durch entsprechende Hinweise ausräumen. Da mit der Erarbeitung ein nicht zu vernachlässigender Aufwand verbunden ist, muss die Entscheidung von einer Nutzenanalyse abhängig gemacht werden. In einem Softwareunternehmen, das eine größere Zahl von Entwicklern schulen möchte, dürfte sich der Gewinn an Klarheit und Zeit durchaus positiv auswirken.

Liste als Gedankenstütze

Unabhängig davon ist festzuhalten, dass für die Anwendung des Regelwerks in einer Anforderungsanalyse die Darstellungen in (Rupp07) nicht geeignet sind. Es mangelt an der nötigen Übersichtlichkeit aufgrund der ausführlichen Beschreibungen und der zahlreichen Beispiele zu den einzelnen Regeln. Daher ist der Einsatz einer Auflistung der Regeln empfehlenswert. Diese sollte die Regelsätze mit den zugehörigen Regelanweisungen beinhalten. Sie kann damit als Gedankenstütze dienen und den Zeitaufwand beim Einsatz des Regelwerks verringern. Dies lässt sich durch eine einfache Rechnung begründen:
Geht man von 10 Anforderungen aus, die mit dem SOPHIST-REgelwerk bearbeitet werden sollen, so muss jede der 25 Regeln für jede Anforderung geprüft werden. Es ergibt sich folglich:

10 Anforderungen x 25 Regeln = 250 Prüfungen

Dass hier nur die Verwendung einer Regelauflistung (oder eines vergleichbaren Hilfsmittels) die Vollständigkeit der Prüfung gewährleisten kann, ist einleuchtend. Dass dies mit einer Liste sehr viel schneller zu bewerkstelligen ist als mit der sich über mehrere Seiten erstreckenden Darstellung in (Rupp07) dürfte ebenfalls einleuchten. Dies wird durch die Erkenntnisse aus der Simulation bestätigt.

Bedarf es eines Hintergrundwissens um das Regelwerk zu verstehen?

Um das SOPHIST-REgelwerk verstehen und anwenden zu können, wird ein umfassendes, grammatikalisches Grundwissen vorausgesetzt. Der Anwender des Regelwerks sollte mit Begriffen wie „Adjektiv", „Verb"... umgehen können. Die verschiedenen, grammatikspezifischen Fachbegriffe werden in (Rupp07) - wie erwähnt - zum Großteil mit Beispielen erläutert, jedoch selten definiert. Besonders bei spezielleren Begriffen wie z. B. „Prozesswort" wirkt sich dies negativ aus, da hier zusätzliche Fachliteratur nötig ist.

Grammatikalisches Grundwissen

Neben dem grammatikalischen Grundwissen werden auch mehrere Begriffe aus der Neuro-Linguistischen-Programmierung als bekannt vorausgesetzt. Hier kommt erschwerend hinzu, dass diese Begriffe kaum in Lexika oder im Internet auffindbar sind und sich die Recherche somit entsprechend schwierig gestaltet.

Weiteres Wissen

4.1.3 Möglichkeit der Verinnerlichung der Regeln

Ist es möglich, die Regeln intuitiv anzuwenden und wie lange dauert es (Anzahl der Iterationen)?

Unter Verinnerlichung bzw. intuitiver Anwendung wird in diesem Zusammenhang verstanden, dass ein Systemanalytiker beim Hören bzw. Lesen einer Anforderung deren Fehler erkennen kann ohne das Regelwerk schrittweise anzuwenden.

Begriffsbestimmung

Das Projektteam konnte die Erfahrung machen, dass dies für wenige Regeln bereits nach der Analyse der ersten Anforderungen in der ersten Iteration möglich war. Die Zahl dieser Regeln stieg bis zum Ende der Simulation stetig an, wobei davon auszugehen ist, dass die intuitive Anwendung des *gesamten* Regelwerks nicht möglich ist.

Zeitdauer

Besonders gut prägen sich häufig anwendbare Tilgungsregeln und die Definitionsregeln ein. Demgegenüber stehen die Generalisierungsregeln, deren Anwendung leichter vergessen wird, sowie sämtliche Regeln, die selten eingesetzt werden.

Unterschied je Regel

Das Projektteam erklärt sich den Unterschied zwischen Tilgungsregeln und Generalisierungsregeln dadurch, dass Tilgungsregeln meist die Spezifikation *eines* Prozesses in einer Anforderung prüfen (z. B. Regel 1, 2, 3, 5, 6), Generalisierungsregeln jedoch oftmals *mehrere* Objekte (z. B. Regel 9, 10, 12, 13, 14, 15, 16). Dadurch wird die Prüfung aufwendiger und weniger intuitiv. Zudem sind fehlende Informationen leichter zu erkennen als ungenaue Informationen. Schließlich treten Generalisierungen und Tilgungen oftmals gemeinsam auf, weil auch Generalisierungsanzeichen getilgt werden. Die dadurch notwendige doppelte Untersuchung ist erst nach intensiverer Beschäftigung mit einer Anforderung möglich.

Ursache

4.1.4 Anwendbarkeit auf Arten der Anforderungserhebung

Kann das SOPHIST-REgelwerk auf mündliche Anforderungen angewendet werden?

Die direkte Anwendung des SOPHIST-REgelwerks bei der Anforderungserhebung erfordert viel Erfahrung im Umgang mit den Regeln. Wenn die Erfassung der Anforderungen in einem mündlichen Interview zwischen Anforderungssteller und –erheber geschieht, ist es für Letzteren schwierig, die Anforderungen gleichzeitig fachlich nachzuvollziehen und auf ihre Regelkonformität zu kontrollieren.

Grundprobleme bei direkter Anwendung

Nach Aussagen von Chris Rupp ist die Regelwerksanwendung durchaus und gerade bei mündlichen Anforderungen gut durchführbar. Sie ordnet die Anwendung des SOPHIST-Regelwerks daher bei den Befragungsmethoden ein.[10] Sie räumt jedoch ebenfalls ein, dass vor einer solchen Anwendung das Regelwerk entsprechend geübt bzw. geschult werden sollte.[11] Das Projektteam vertritt demgegenüber die Meinung, dass selbst mit Schulung und Erfahrung die Anwendung des *kompletten* Regelwerks bei der Erhebung auf Grund des Umfangs und der Komplexität nicht möglich ist. Diese Ansicht wird dadurch bestätigt, dass die SOPHIST-Group nach eigenen Angaben kein Training für die interaktive Anwendung des Regelwerks anbietet.

Anwendbarkeit nach Chris Rupp

[10] Vgl. (Rupp07) Abb. 5.1 S. 110
[11] Vgl. (Rupp07) S. 123, S. 147, S. 317

Als möglicher Lösungsweg wird vorgeschlagen, den größtmöglichen Teil des Regelwerks sofort bei der Anforderungserhebung anzuwenden und dabei die Informationen für die Korrektur der Anforderungen nachzuerheben und mitzuprotokollieren. Dadurch kann erheblich Zeit in der Nachbearbeitung der Anforderungen eingespart werden, da keine eventuell falschen Mutmaßungen angestellt werden müssen, die bei einer Regelanwendung ohne Einbezug des Anforderungsstellers erfolgen. Hiermit kann bereits zu Beginn ein qualitativ hochwertiger Stand der Anforderungen erzielt werden.

Vor- und Nachteile einer mündlichen Anwendung

Demgegenüber steht ein größerer Zeitaufwand im Erhebungsgespräch, der die Motivation und Akzeptanz des Anforderungsstellers gefährden kann. Abgesehen davon, sollte bedacht werden, dass mündliche Anforderungen im Normalfall eine schlechtere Qualität haben als schriftliche Anforderungen, die vom Anforderungssteller nochmals nachkontrolliert wurden. Da, wie später beschrieben, die Anwendung des Regelwerks sehr zeitaufwändig ist, erscheint eine möglichst gute Anforderungsbasis aber wichtig.

4.2 Eindeutigkeit der Regeln

Haben die Anwender des SOPHIST-REgelwerks dieselben Fehler entdeckt bezogen auf einzelne Anforderungen (bei mehrfacher Versuchsausführung)?

Im ersten Iterationsschritt der Simulation führten die Projektteammitglieder jeweils eine eigenständige Analyse der Anforderungen durch. Das Ziel dieser Vorgehensweise war es herauszufinden, ob sie zu den gleichen Ergebnissen kommen würden. Durch den Grad der Übereinstimmung können Rückschlüsse auf die Erkennbarkeit der Regelanwendung und die eindeutige Formulierung von Regelanweisungen gezogen werden.

Getrennte Anwendung des Regelwerks

Die Ergebnisse der einzelnen Teammitglieder wurden mit Hilfe einer Excel-Tabelle ausgewertet. Übereinstimmungen konnten bei den Regeln entdeckt werden, die nur wenig Raum für Interpretationen ließen.

Übereinstimmung

Dies ist auch auf die große Anzahl bestimmter Fehler in den Ausgangsanforderungen zurückzuführen. So waren die Anforderungen bspw. fast ausschließliche im Passiv formuliert, so dass Regel 1 zur Aktivumformulierung regelmäßig eingesetzt werden konnte.

Dort wo es Abweichungen in der Analyse gab, ist dies auf die unterschiedliche Interpretation einzelner Regeln zurückzuführen. Es konnten in der späteren Diskussion der Ergebnisse die Sichtweisen jedoch erläutert werden, so dass auch für diese eine große Übereinstimmung erzielt werden konnte.

Abweichungen

4.3 Methodisches Vorgehen

Die Analyse einer Anforderung ist ein häufig wiederholter, aufwändiger und gleichartiger Prozess während der Systemanalyse. Da in Softwareprojekten schnell mehrere hundert Anforderungen entstehen, ist es wünschenswert, diesen Prozess möglichst rationell und zielgerichtet zu gestalten. Nachfolgend soll er daraufhin untersucht werden.

4.3.1 Reihenfolge der Regeln

Gibt es eine sinnvolle Reihenfolge der Anwendung / Abhängigkeiten zwischen den Regeln?

Die Ordnung der Regeln im SOPHIST-Regelwerk in (Rupp07) wurde aus der willkürlichen Einteilung des Neuro-Linguistischen-Programmierens in die Bereiche Tilgung, Generalisierung, Verzerrung übernommen. Dies ist die einzige erkennbare Gliederung, der das Regelwerk folgt. Innerhalb der Bereiche gibt es keine Ordnung der Regeln etwa nach der Reihenfolge der Anwendung oder eventuellen Abhängigkeiten.

Ordnung der Regeln in (Rupp07)

Bei der Anwendung stellt sich die Frage, mit welcher Regel zu beginnen ist und wie die Reihenfolge aussieht. Hierfür gibt es in (Rupp07) keinerlei Vorgaben. Dem Anwender wird somit erst in der Anforderungsanalyse klar, welche Regeln in Beziehung stehen und welche Regeln eine erneute Anwendung einer anderen mit sich bringen. Automatisch geht der Anwender somit zunächst nach der Reihenfolge vor, in der die Regeln niedergeschrieben wurden vor, was häufig einen Mehraufwand bedeutet. Eine Richtlinie über eine mögliche Reihenfolge wäre hier eine sinnvolle Unterstützung.

Probleme bei der ersten Anwendung

Eine frühere Auflage (Rupp02) enthält zur Unterstützung einen Algorithmus, der eine Reihenfolge vorgibt. Der Vorteil liegt darin, dass durch die Berücksichtigung von Abhängigkeiten die doppelte Anwendung von Regeln vermieden werden kann. Nach Meinung des Projektteams gewährleistet dieser Algorithmus dies jedoch nicht. In Kap. 8 befindet sich daher ein selbst erstell-

Leitfaden für das Vorgehen

ter Leitfaden zur Anwendung, der nach Meinung des Projektteams besser geeignet ist.

4.3.2 Anzahl der Iterationen

Nach wie vielen Iterationen werden keine Fehler mit dem Regelwerk mehr gefunden?

Überprüfungsvorgang

In der Simulation kam es zur zweimaligen Anwendung der Regeln und Nachfragen beim Anforderungssteller, bis keine Fehler mehr gefunden werden konnten. Dabei wurde das Regelwerk ohne Ausnahme durchgeführt und jede Anforderung, teils durch Nachfragen beim Anforderungssteller teils durch eigenverantwortliche Tätigkeiten, in einen regelkonformen Zustand versetzt.

Gründe für Iterationsanzahl

Gründe, warum bei der Überprüfung zwei Iterationen benötigt wurden, werden hier im Folgenden aufgeführt. Zunächst müssen Umformulierungen, die aus der ersten Anwendung entstanden sind, mit dem Anforderungssteller abgeglichen werden. Häufig erwachsen daraus neue Anforderungen, die ihrerseits auch eine Prüfung durch das Regelwerk zu erfahren haben. Da ein vollständiges Anwenden des Regelwerks nicht direkt bei der Erfassung möglich ist, wird eine weitere Iteration benötigt. Die meisten neu entstandenen bzw. zuerst vergessenen Anforderungen werden bei dieser ersten Rücksprache aufgedeckt. Die Entstehung neuer Anforderungen kann vom Anwender des Regelwerks beeinflusst werden. Er kann durch eigene Annahmen dem Anforderungssteller vorgreifen und muss nicht für triviale, durch die Anwendung aufkommende Fragen mit ihm Rücksprache halten. Die durch dieses eigenmächtige Handeln neu entstandenen Anforderungen müssen dann vom Anforderungssteller abgesegnet werden. Daher bedingen getroffene Annahmen, obwohl sie Zeit und Durchläufe sparen, in jedem Fall mindestens eine zusätzliche Iteration.

4.3.3 Gefundene Fehler (mit Kategorie) und Rückfragen

Wie viele Fehler und welche Arten von Fehlern werden mit dem SOPHIST-REgelwerk entdeckt?

Der Verlauf der Anforderungsanalyse und die entdeckten Fehler wurden bereits ausführlich in Kap. 3.5 und 3.7 dargestellt. An dieser Stelle wird daher nur eine Zusammenfassung der gefundenen Fehler gegeben.

Zahl der Fehler und Rückfragen

Während den Iterationen in der Analyse der 19 Anforderungen (plus fünf Definitionen) wurden die Regeln ca. 54 Mal eingesetzt, um die Anforderungen umzuformulieren und ca. 94 Mal für das Erstellen von Rückfragen. Aus Letzteren entstanden 124 Fragen an den Anforderungssteller.
Geht man davon aus, dass mit jeder Regelanwendung bzw. Rückfrage ein Fehler bzw. eine fehlende Information aufgedeckt werden konnte, ergibt sich daraus ein Verhältnis von über neun Fehlern pro Anforderung.

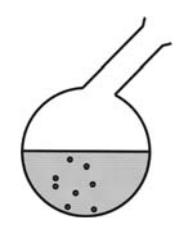
Fehlerkategorien

Fasst man die auftretenden Fehlerarten in Kategorien zusammen und bewertet man die einzelnen Anforderungen nach diesen Kategorien, ergibt sich das in der nachfolgenden Tabelle dargestellte Bild an auftretenden und entdeckten Fehlerarten. Auch wenn diese Zahlen von verschiedenen Faktoren (siehe nachfolgende Hinweise) und der Formulierung der Fragen abhängen, verdeutlichen sie dennoch, dass mit dem Regelwerk eine Vielzahl an Fehlern entdeckt werden kann.

Fehlerkategorie	**Zahl der Fehler (in 19 Anforderungen)**
Nicht alle beteiligten Objekte angegeben	18
Umsetzung der Funktionalität fehlt oder ist unpräzise	18
Beteiligte Objekte werden unpräzise spezifiziert	9
Sonderfälle fehlen	8
Implizite Anforderungen existieren	8
Floskeln oder unnötige Satzbestandteile existieren	8
Redundante Anforderung	2
Funktionalität vergessen	1

Tabelle 2: Gefundene Fehlerarten und -anzahl

Bei der Bewertung der gefundenen Fehler stellt sich im Besonderen die Frage, inwieweit die Fehler repräsentativ sind. Hier lassen sich verschiedene, teilweise bereits erwähnte Einflußfaktoren zusammenfassen:



Anforderungssteller

- ◇ Erhebungstechnik (schriftlich, mündlich)
- ◇ Sprachgebrauch/-kompetenz des Anforderungsstellers
- ◇ Anzahl der Anforderungssteller und deren Vorstellungen vom System (Widersprüche)
- ◇ Gründlichkeit des Anforderungsstellers (aktive oder passive Rolle bei der Verbesserung der Anforderungen)

System

- ◇ Umfang und Komplexität des Systems
- ◇ Umfang des Anforderungsdokuments
- ◇ Detaillierung des Anforderungsdokuments (gewünschte Freiheitsgrade in der Spezifikation)

Projektteam

- ◇ Interpretation des Regelwerks
- ◇ Reihenfolge der Regelanwendung
- ◇ Gründlichkeit in Anwendung und Dokumentation

Simulations-szenario

Die Zahl der Einflussfaktoren und deren mögliche Ausprägungen zeigen, dass die Simulation nur für das dortige Szenario der Anforderungsanalyse repräsentativ sein kann - in diesem Fall:

> „Ein Anforderungsteller spezifiziert grob und mündlich ein eher einfaches System. Er nimmt an der Verbesserung passiv teil und überlässt diese überwiegend dem Projektteam."

Das Projektteam geht jedoch davon aus, dass genau dieses Szenario einerseits Praxisbezug besitzt und andererseits die Ergebnisse für dieses Szenario repräsentativ sind.

4.3.4 Entdeckte Neuanforderungen

Konnten neue Anforderungen durch die Anwendung des SOPHIST-REgelwerks entdeckt werden?

Neue Anforderungen

Durch die Anwendung des SOPHIST-REgelwerks konnten im Verlauf der Simulation neue Anforderungen aufgedeckt werden, die im ersten Gespräch mit dem Anforderungssteller nicht erwähnt wurden und die komplett neue Sachverhalte beschrieben. Hierunter fällt bspw. die Abfrage sämtlicher reisespezifischer Daten wie z. B. Reiseziel oder Verpflegungsart vom CRS-Server. Diese Anforderung konnte durch die Anwendung von Regel 7 aufgedeckt werden. Sie zeigt exemplarisch die Fähigkeit des Regelwerks, implizite Anforderungen zu erkennen.

4.3.5 Auswirkung auf spätere Anforderungsänderungen

Wie beeinflusst das Regelwerk spätere Änderungen der Anforderungen?

Kunstwerkeffekte

Da, wie später beschrieben, der Arbeitsaufwand selbst bei ausreichender Erfahrung mit dem Regelwerk sehr hoch und die Bearbeitung teilweise sehr anstrengend ist, kann vermutet werden, dass Systemanalytiker Änderungen an den überarbeiteten Anforderungen negativ gegenüberstehen. Das Projektteam kann diese These mit der Erfahrung aus der Simulation untermauern. Inwieweit dieser Effekt in der Praxis von Bedeutung ist bleibt fraglich. Es kann davon ausgegangen werden, dass in einem professionellem Umfeld derartige Effekte keine Rolle spielen.

Änderungsaufwand

Ferner ist davon auszugehen, dass der Aufwand für spätere Änderungen an den Anforderungen durch die Anwendung des SOPHIST-REgelwerks sinkt. Die ursprünglichen Anforderungen sind präziser und übersichtlicher in ihrer Gestaltung, Formulierung und Darstellung, so dass es leichter wird, Änderungen aufzunehmen und einzupflegen. Zudem kann der Systemanalytiker durch die intensive Beschäftigung mit den Anforderungen, besser die Auswirkung von Anforderungen beurteilen.

4.4 Vollständigkeit des Regelwerks

Um die Vollständigkeit des Regelwerks untersuchen zu können, sind zunächst einige Vorüberlegungen nötig.

Im Requirements Engineering gibt es verschiedene Tätigkeiten, die im Rahmen einer Systementwicklung durchgeführt werden müssen. So ist nicht nur die Erhebung von Anforderungen eine zu erledigende Aufgabe, sondern u. a. auch deren Dokumentation oder das Erstellen eines Analysemodells. Zudem verfolgen diese Tätigkeiten verschiedene Ziele. Anforderungen und Analysemodell sollen bspw. eindeutig, widerspruchsfrei und vollständig sein. Die Bewertung der Vollständigkeit des Regelwerks fällt daher unterschiedlich aus, je nachdem, welche Aufgaben und welche Ziele der Bewertung zu Grunde gelegt werden.

Aufgaben im Requirements Engineering

An dieser Stelle erfolgt rein die Betrachtung der Anforderungserhebung für die das Regelwerk ausgelegt ist, auch wenn die Erstellung des Analysemodells und mehr noch die Anforderungsdokumentation stark von der Erhebung abhängen und nicht immer getrennt werden können. Als Ziel der Anforderungserhebung werden gemäß der Festlegung der SOPHIST-Group eindeutige, vollständige und widerspruchsfreie Anforderungen betrachtet.[12]

Zielsetzungen

4.4.1 Fehlen und Notwendigkeit der Regeln

Können fehlende Regeln ausgemacht werden?

Inwieweit Regeln fehlen kann aus den Hauptzielen der Regeln hergeleitet werden. Bei der Betrachtung lässt sich unterscheiden, ob das verfolgte Ziel mehr für eine einzelne Anforderung oder das gesamte Anforderungsdokument gilt. Nachfolgende Tabelle zeigt den Hauptbeitrag jeder Regel zu den oben genannten Zielen.

Ebene	Vollständigkeit	Eindeutigkeit	Widerspruchs-freiheit
Anforderung einzeln	1, 2, 3, 4, 5, 6, 11, 17	2, 9, 10, 13, 14, 15, 16, 18, 19, 20, 21, 22	
Anforderungen gesamt	7, 8	12, 23, 24, 25	

Tabelle 3: Beitrag der Regeln zu den Zielen

Aus dieser Tabelle sind zunächst drei Erkenntnisse abzuleiten:

- Die Eindeutigkeit und Vollständigkeit einzelner Anforderungen wird mit vielen Regeln abgedeckt. Diese Bereiche werden als große Stärke des

[12] Vgl. (SOPHIST01), S. 4 f.

Regelwerks angesehen. Den Bereich der Eindeutigkeit unterstützen vor allem Glossareinträge, definierte quantitative Angaben, der Verzicht auf Nebensätze und die Untersuchung nach Substantiven ohne Bezugsindex. Durch sie wird sowohl die Mehrdeutigkeit auf Wortebene (lexikalisch), wie auch innerhalb von Sätzen (strukturell, semantisch) bzw. zwischen Sätzen (pragmatisch) ohne großen Aufwand bis zu einem gewissen Grad verhindert.

◇ Das Regelwerk prüft mit keiner Regel auf Widersprüche innerhalb einer Anforderung bzw. zwischen Anforderungen. Zu erklären dürfte dies dadurch sein, dass Widersprüche innerhalb von Anforderungen auf der einen Seite eine eher untergeordnete Rolle in der Praxis haben dürften. Widersprüche zwischen *natürlichsprachlichen* Anforderungen dürften auf der anderen Seite schwer und nur unter großem Aufwand erkennbar sein. Eine Verlagerung der Prüfung auf den späteren Teil der Systemanalyse, in dem ein Analysemodell erstellt wird, erscheint daher angemessen.

◇ Das Regelwerk konzentriert sich mehr auf einzelne Anforderungen als auf den gesamten Anforderungskatalog. Im Bereich der Vollständigkeit wird dies dadurch begründet, dass nur durch Analysemodelle oder zusätzliche Erhebungstechniken die Vollkommenheit sicherzustellen ist.[13] Für den Bereich der Widerspruchsfreiheit wurde bereits eine ähnliche Annahme getätigt. Die Eindeutigkeit zwischen Anforderungen dürfte nach Meinung des Projektteams jedoch mit wenig Aufwand erreichbar sein. Hier wurden auch während der Simulation fehlende Regeln erkannt, die bspw. klarstellen sollten, ob bzw. wie sehr Anforderungen kontext-unabhängig sein müssen. Auch wurden Regeln vermisst, die den Umgang mit stark zusammenhängenden Anforderungen (Anwendungsfälle, zeitliche Abfolgen, Anforderungen zu einer Bildschirmmaske etc.) klären. Bei diesen resultieren nach Meinung des Projektteams zu umständliche Anforderungen aus dem Regelwerk, die das Verständnis unnötig erschweren.

---

[13] Vgl. (Rupp07), S. 155

Wurden alle Regeln angewendet?

Nahezu vollständige Anwendung

Bis auf Regel 18 wurden alle Regeln entweder bei der Analyse oder bei der Neuformulierung angewendet. Einige Regeln wurden dabei besonders oft angewendet (z. B. Regel 1, 3, 5, 9), andere äußerst selten (z. B. Regel 2, 4). Aus der geringen oder fehlenden Häufigkeit einer Anwendung kann jedoch nicht geschlossen werden, dass die Regel nicht notwendig ist. Einerseits hat das Anforderungsdokument Einfluss auf die Häufigkeit, andererseits ist eine Regel auch dann sinnvoll, wenn sie nur einen geringen Anteil der Anforderungen verbessert (z. B. Regel 4).

Unnötige Redundanz

Als überflüssig können Regeln damit nur angesehen werden, wenn sich ihr Inhalt mit anderen Regeln deckt oder wenn durch ihr Zusammenführen mit anderen Regeln eine einfachere Anwendbarkeit des Regelwerks entsteht. Dies ist in einigen Fällen wie in Kap. 3.7 erwähnt gegeben. An dieser Stelle sollen daher die dortigen Vorschläge nochmals zusammengefasst werden:

◇ Regel 17 wird durch Regel 2 abgedeckt und kann entfernt werden.
◇ Regel 10, 14, 15 und 16 beziehen sich stark aufeinander und sollten zu einer Regel zusammengefasst werden.
◇ Regel 19 und 20 sind nahezu bedeutungsgleich und sollten zu einer Regel zusammengefasst werden.

4.4.2 Nicht-funktionale Anforderungen

Lässt sich das SOPHIST-REgelwerk auch auf nicht-funktionale Anforderungen anwenden?

Unter der Bezeichnung „nicht-funktionale Anforderungen" werden verschiedene Arten von Anforderungen zusammengefasst. In (Rupp07) umfassen nicht-funktionale Anforderungen technische Anforderungen, Anforderungen an die Benutzerschnittstelle, Qualitätsanforderungen, Anforderungen an sonstige Lieferbestandteile, Anforderungen an durchzuführende Tätigkeiten und rechtlich-vertragliche Anforderungen.[14]

Keine Eignung

Obwohl in der Simulation nur Beispiele für die Teilbereiche „Anforderungen an die Benutzerschnittstelle" und „Technische Anforderungen" auftauchten, kann davon ausgegangen werden, dass das Regelwerk für nicht-funktionale Anforderungen allgemein nicht geeignet ist. Zwar können einige Grundideen des Regelwerks die Qualität von vielerlei Texten verbessern, jedoch ist es zu stark auf die Schilderung von Vorgängen und Funktionalitäten spezialisiert, so dass viele Regeln bei nicht-funktionalen Anforderungen nicht notwendig sind.

Umgang mit nicht-funktionalen Anforderungen

Im Zusammenhang mit der Fragestellung erscheint interessanter, wie der Umgang mit den nicht-funktionalen Anforderungen während der Simulationsumgebung gelang.
Hier kann zunächst festgestellt werden, dass das Regelwerk technische Anforderungen nur unter bestimmten Voraussetzungen zulässt. Diese Aussage taucht in der Erläuterung zu Regel 5 auf.[15] Zählt man Spezifikationen von technisch bedingten Fehlerbehandlungen nicht zu technischen Anforderungen, war diese Einschränkung gut durchführbar. Bspw. der Bedarf Übertragungsprotokolle für die Kommunikation mit dem CRS-Server zu spezifizieren oder einen Datenbankhersteller für die geforderte Kundendatenbank festzulegen, erschien für das Formulieren kompletter Anforderungen nicht nötig. Die Anforderungen bleiben weiterhin verständlich, was es ermöglicht,

---

[14] Vgl. (Rupp07), S. 256
[15] Vgl. (Rupp07), S. 152

die technischen Anforderungen auf einen eigenen Teil eines Anforderungsdokuments zu verlagern.
Anders war es bei Anforderungen zur Benutzerschnittstelle. Hier fand nach Meinung des Projektteams teilweise eine Vermischung statt. Die Trennung von Funktionalität und Benutzeroberfläche war oftmals nicht möglich, wenn weiterhin vollständige Anforderungen verfasst werden sollten. Demgegenüber reichen die Angaben in den Anforderungen für eine komplette Spezifikation der Benutzeroberfläche nicht aus. Sie sind zudem wenig übersichtlich und verursachen großen Aufwand in der Formulierung. Der Einsatz von Tabellen oder Grafiken als Ergänzung zu den Anforderungen könnte hier Abhilfe schaffen. In (Rupp07) wird auf die Gefahren bei deren Einsatz hingewiesen und gleichzeitig ihre Notwendigkeit festgestellt. Eine brauchbare Lösung des Problems wird nicht gegeben. Hier erscheinen weitere Empfehlungen notwendig, die zeigen, wie funktionale Anforderungen und Anforderungen an die Benutzeroberfläche übersichtlich, eindeutig und redundanzfrei kombiniert werden können.

4.5 Zeitlicher Aufwand für die Anwendung

Wie lange ist die Bearbeitungsdauer einer Anforderung?

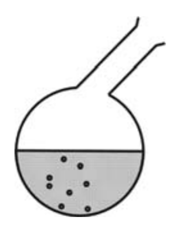
Analyse-schritte

Aus dem Vorgehensmodell in Kap. 3.3 kann die Bearbeitungszeit von erhobenen Anforderungen in einer Iteration in die Schritte

- ◇ Vorbereitung der Rückfrage,
- ◇ Durchführung der Rückfrage,
- ◇ Nachbearbeitung der Rückfrage und
- ◇ Verabschiedung der Anforderungen

unterteilt werden. Nachfolgend wird der Zeitaufwand für die einzelnen Teile gezeigt, der sich in der Simulation ergeben hat. Da der Großteil der Anforderungen in der ersten Iteration verabschiedet werden konnte, beschränkt sich die Darstellung hierauf. Die Ermittlung der Durchschnittswerte geht wiederum von 19 anfänglichen Anforderungen aus, wozu es die als Glossareinträge eingestuften Anforderungen nicht mitzählt.

Es ist unzweifelhaft, dass der ermittelte Zeitaufwand von der Simulationsumgebung beeinflusst ist. Es wird davon ausgegangen, dass er durch die geringe Erfahrung des



Projektteams in der Anwendung des Regelwerks und durch die Qualität mündlich aufgenommener Anforderungen nach oben beeinflusst wurde. Auf der anderen Seite dürfte unter realen Bedingungen (größeres System, mehrere Anforderungssteller) mehr Zeitaufwand für die Suche nach impliziten Anforderungen, das Erstellen von allgemein anerkannten Glossareinträgen o. ä. anfallen.
Die Beeinflußung der Anwendung, durch die für diese Bewertung notwendige Dokumentation, wird als gering eingeschätzt.

Vorbereitung

Im Vorbereitungsteil wurden die Anforderungen mit dem Regelwerk analysiert, umformuliert und Rückfragen dokumentiert. In der Simulation ergab sich für diese Tätigkeiten ein durchschnittlicher Zeitaufwand von 18 Minuten pro Anforderung, wobei das Ermitteln der anwendbaren Regeln etwa halb so viel Zeit benötigte wie das Umformulieren der Anforderungen bzw. das Formulieren der Fragen.

Durchführung

Im Durchführungsteil wurden die Rückfragen an den Anforderungssteller gestellt und dessen Antworten dokumentiert. Hier reichten etwa sechs Minuten pro Anforderung aus.

Nachbearbeitung

In der Nachbearbeitung wurden die ermittelten Antworten dann in die ursprünglichen Anforderungen eingearbeitet, wobei oftmals eine Aufteilung in mehrere Anforderungen vorgenommen wurde. Ebenso wurden neue Anforderungen und Glossareinträge formuliert. Pro anfänglicher Anforderung entstand hier ein Aufwand von ca. 40 Minuten, wobei darauf hinzuweisen ist, dass sich in diesem Schritt die Zahl der Anforderungen mehr als verdreifacht hat. Der Aufwand pro neuer Anforderung ist demnach im Bereich von etwa zehn Minuten.

Verabschiedung

In der Validation werden die Anforderungen abschließend begutachtet und verabschiedet. Wie erwähnt ergaben sich hier nur noch wenige Änderungswünsche. Der Zeitaufwand pro anfänglicher Anforderung kann hier mit drei Minuten angegeben werden, bezogen auf die Zahl der neuen Anforderungen lag er unter einer Minute.

Gesamtdauer

Im Gesamten ergibt sich damit eine durchschnittliche Bearbeitungsdauer pro anfänglicher Anforderung von etwa 67 Minuten.

Obwohl die Zeitangaben nur als Orientierungshilfe für den anfallenden Aufwand gesehen werden können, lassen sich doch einige Erkenntnisse über das Regelwerk ableiten:

Erkenntnisse

- ◇ Der große Aufwand in der Vorbereitung zeigt, dass die Anwendung der 25 Regeln viel Zeit in Anspruch nimmt.
- ◇ Der geringe Aufwand für Durchführung und Verabschiedung zeigt, dass der Anforderungssteller durch das Regelwerk eher gering beansprucht wird und zielgerichtete Nachfragen formuliert werden konnten.
- ◇ Der große Aufwand in der Nachbearbeitung ist dadurch begründet, dass hier Anforderungen neu formuliert, wiederum analysiert und ggf. korrigiert werden müssen. Er verdeutlicht, dass eine Trennung von Durchführung und Nachbearbeitung durchaus sinnvoll ist und den Anforderungssteller erheblich entlasten kann.
- ◇ Der große Gesamtaufwand deutet auf ein sehr aufwendiges Verfahren hin.

4.6 Motivation der Simulationsteilnehmer

In der Anforderungserhebung und in der Analysephase gibt es häufig große Probleme durch das mangelnde Interesse seitens der Anforderungssteller. Deshalb soll an dieser Stelle die Motivation der Beteiligten in der Simulation betrachtet werden. Des Weiteren sollen Überlegungen aufgestellt werden, wie bzw. durch welche Maßnahmen die Beteiligten motiviert werden können.

Wie wurde die Motivation der Simulationsteilnehmer über den Verlauf der Simulation vom Projektteam aufgefasst?

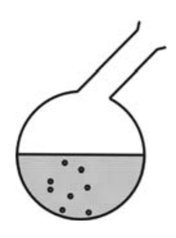
Motivation des Anforderungsstellers

Der Anforderungssteller für das Reisebuchungssystem war trotz seiner hohen Belastung durch sein Studium sehr motiviert. Er zeigte großes Interesse an den Veränderungen, die sich durch die Anwendung des SOPHIST-REgelwerks ergeben hatten. Auch die Rückfragen nahm er gern auf und beantwortet diese ausführlich und präzise. Er zeigte sich positiv überrascht über die Qualität und die Quantität der „neu entstandenen" Anforderungen. Die Verbesserung der Anforderungen wurde nicht als „Besserwisserei" erachtet, sondern honoriert.

Motivation des Projektteams

Die Motivation des Projektteams wurde zum einen durch den hohen Bearbeitungsaufwand beeinflusst. Zum anderen ergaben sich bei der Interpretation der Regeln immer wieder Probleme und Diskussionen, woraus sich anfangs eine Missstimmung im Team ergab, da nicht befriedigend geklärt werden konnte, wie die einzelnen Regeln tatsächlich auszulegen sind oder wie detailliert sie angewendet werden sollten.
Eine weitere Schwierigkeit, die sich bei der Anwendung des SOPHIST-REgelwerks zeigte, war der schnelle Konzentrationsverlust beim Einsatz. Aufgrund der oft eintönigen aber anspruchsvollen Änderungen (z.B. von Passiv nach Aktiv, ...) konnte bereits nach wenigen Anforderungen ein deutliches Nachlassen der Konzentration beobachtet werden.

In der Praxis ist davon auszugehen, dass die Motivation des Anforderungsstellers sehr viel geringer ausfallen wird als in der Simulation. Zwar war die Teilnahme an der Simulation für den Anforderungssteller auch nur eine Nebentätigkeit, aber er hatte sich freiwillig dazu bereiterklärt.



Anforderungssteller in der Praxis nehmen nur selten freiwillig an einer Anforderungsermittlung teil und sehen sie als eine Störung in ihrer täglichen Arbeit an.[16] Für den Fall, dass sie sich intensiv mit einem Anforderungsdokument beschäftigt haben, ist die Gefahr auch größer, dass sie Nachfragen und Korrekturen als Beleidigung ihrer Kompetenz einstufen.

Wie kann die Motivation der Teilnehmer beeinflusst werden?

Da die Motivation aller Teilnehmer entscheidend für den Erfolg ist, sollte hierauf besonders geachtet werden. Im Folgenden wird aufgezeigt, wie die Motivation beider Seiten gesteigert bzw. aufrechterhalten werden kann.

Einbeziehen des Anforderungsstellers

Die Motivation der Anforderungssteller kann dadurch gesteigert werden, dass sie in die Anwendung des SOPHIST-REgelwerks miteinbezogen werden. Hierunter ist zu verstehen, dass das SOPHIST-REgelwerk den Anforderungsstellern erläutert wird und die Notwendigkeit einer qualitativen Verbesserung von Anforderungen vermittelt wird. Des Weiteren sollte den An-

[16] Vgl. (Robertson/Robertson06), o. S.

forderungsstellern deutlich aufgezeigt werden, wie sich die Anforderungen verändert haben, denn die Anforderungen werden bereits nach der ersten Anwendung des SOPHIST-REgelwerks nur noch wenige Ähnlichkeiten mit den Originalanforderungen aufweisen. Es kann dabei das Problem auftreten, dass sich Anforderungssteller missverstanden fühlen und fälschlicherweise schließen, dass die von ihnen formulierten Anforderungen eigentlich nicht benötigt wurden. Hier kann es hilfreich sein, dass immer die Originalanforderungen sowie die neuen Anforderungen parallel vorgestellt werden. Anforderungssteller erhalten damit einen besseren Überblick und können den Prozess leichter nachvollziehen.

Pausen zur Konzentrationserhaltung

Für die Motivation der Anwender des SOPHIST-REgelwerks sind ausreichende Pausen nützlich und nötig. Dadurch wird die Monotonie des Regelwerks aufgelockert und gewährleistet, dass entscheidende Umformulierungen aufgrund eines Konzentrationsmangels nicht übersehen werden. Auch die Festlegung der Interpretation der Regeln und des Detaillierungsgrades vor dem Einsatz wirken motivierend, da so Doppelarbeit und unnötige Diskussionen erspart werden.

Toolunterstützung

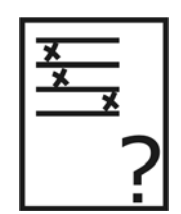
Eine Erleichterung der Anwendung durch ein Unterstützungstool, das bspw. Fragen zu den Regeln vorgibt, würde die Arbeit weiter vereinfachen und somit ebenfalls die Motivation und die Konzentration fördern.

4.7 Das Vorgehen aus der Sicht des Anforderungsstellers

Im folgenden Kapitel soll auf die Bewertung des Vorgehens aus der Sicht des Simulationssteilnehmers eingegangen werden. Hierzu wurde dem Anforderungssteller ein Fragebogen ausgehändigt.

4.7.1 Zeitaufwand

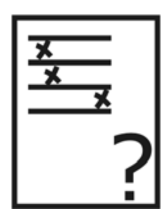
Zufriedenheit mit Zeitaufwand



Im Bezug auf den benötigten Zeitaufwand äußerte sich der Anforderungssteller positiv. Er empfand sowohl die Dauer der ersten Erhebung als auch den Zeitaufwand für die Rückfragen in den einzelnen Iterationsschritten als angemessen und vertretbar.

Der Anforderungssteller begründete seine Zufriedenheit damit, dass er sicher war, dass das Projektteam die von ihm gestellten Anforderungen und Anmerkungen auch verstanden hatte.

4.7.2 Verständlichkeit



Im Zusammenhang mit der Verständlichkeit wurde der Anforderungssteller gefragt, inwieweit er zum einen die Umformulierungen und zum anderen die neuen Anforderungen nachvollziehen konnte.

Umformulierungen verständlich

Hier gab der Anforderungssteller an, dass er die Umformulierungen auch ohne vorherige Kenntnis des SOPHIST-REgelwerks verstehen und nachvollziehen konnte. Es kann gefolgert werden, dass die Umformulierungen begründet sind und sie auf Fehler und Mängel bei den Anforderungen hinweisen. Wäre dies nicht der Fall, so wären dem Anforderungssteller sicherlich einige Umformulierungen aufgefallen, die seiner Meinung nach überflüssig, unnötig oder falsch gewesen wären.

Neue Anforderungen verständlich

Mit den umformulierten Anforderungen ergaben sich ebenfalls nur selten Verständnisprobleme. Dort, wo welche auftraten, kann dies auf die hohe Komplexität einzelner Anforderungen – wie zum Beispiel Anforderung Nr. 23 oder Nr. 28 – zurückgeführt werden.

4.7.3 Akzeptanz



Mit den Fragen bezüglich der Akzeptanz sollte festgestellt werden, ob der Anforderungssteller die Anwendung des SOPHIST-REgelwerks als akzeptabel empfunden hat oder ob er den Aufwand als nicht gerechtfertigt ansieht.

Aufwand akzeptabel im Vergleich zum Ergebnis

Es konnte festgestellt werden, dass die zeitliche Belastung aller Iterationsschritte vom Anforderungssteller akzeptiert wurde. Der Teilnehmer sieht den Aufwand seinerseits gemessen am erzielten Ergebnis als gering an. Es kann geschlussfolgert werden, dass der Anforderungssteller den Aufwand akzeptiert, da er die Verbesserungen der Anforderungen erkennen kann und sie für ihn sinnvoll scheinen.

Der Anforderungssteller gibt außerdem an, dass er das SOPHIST-REgelwerk zukünftig auch in eigenen Projekten einsetzten will. Er erkannte, dass durch die Anwendung des SOPHIST-REgelwerks „[...] sichergestellt wird, dass alles systematisch aufeinander aufbaut."

4.8 Zusammenfassung

Zum Abschluss der Bewertung des Vorgehens soll das SOPHIST-REgelwerk nun den eingangs gestellten Anforderungen an eine Methode zur Anforderungsanalyse gegenübergestellt werden.

Erlernbarkeit und Eindeutigkeit

Die Darstellung des Regelwerks in (Rupp07) wird als schlecht eingestuft, worunter die Erlernbarkeit der Theorie stark leidet. Dies ist umso mehr ärgerlich, da das Regelwerk vergleichsweise „einfach" ist und damit bei sorgfältigerer Darstellung eine bessere Erlernbarkeit zu erwarten wäre. In der praktischen Anwendung kann das Regelwerk schnell eingeübt werden, jedoch nicht bis zu dem Maße, dass es komplett und interaktiv in einem Interview eingesetzt werden kann.

Die Eindeutigkeit der Ergebnisse bei Anwendung durch mehrere Personen ist großteils gegeben, jedoch nur, falls sich diese über die Interpretation aller Regeln verständigt haben. Dies spricht ebenfalls für die Erlernbarkeit in der Praxis. Außerdem zeigt es, dass das Regelwerk anwendbar ist, da die Fehler in Anforderungen erkannt werden.

Vollständigkeit bei Minimalität

Das Regelwerk kann als ausreichend vollständig bezüglich der Analyse einzelner Anforderungen angesehen werden. Die Vielzahl der gefundenen Fehler und der neuen Anforderungen untermauern dies.

Sinnvoll erscheinen weitere Regeln, die Abhängigkeiten zwischen Anforderungen behandeln und die Ergänzung einiger Regelanweisungen um Aussagen, die nur in (Rupp07) auftauchen.

Als gute Lösung wird angesehen, dass sich das Regelwerk auf die häufigsten Fälle der Mehrdeutigkeit konzentriert und das Erkennen von Widersprüchen auf den späteren Verlauf der Systemanalyse verschiebt. Weitere Regeln oder normative Vorgaben würden hier einen zu hohen Mehraufwand bedeuten.

Der Inhalt keiner Regel wird als überflüssig angesehen. Es sind jedoch an wenigen Stellen Redundanzen im Regelwerk zu erkennen, außerdem sollten einige Regeln zusammengefasst werden.

Methodisches Vorgehen

Der Systemanalytiker wird durch 25 Regeln gelenkt, die Anweisungen bzw. Fragen für die Anforderungsanalyse vorgeben. Verbesserungswürdig wären die Formulierung vereinzelter Regelanweisungen und die Vorgabe einer Reihenfolge. Sie sollte den Aufwand für die Anwendung minimieren und inhaltliche Abhängigkeiten zwischen Regeln berücksichtigen.
Als ungeeignet und teilweise sogar irreführend haben sich die Signalwörter für die Anwendbarkeit der Regeln erwiesen. Durch sie werden Gesetzmäßigkeiten suggeriert, die nach Meinung des Projektteams in Anforderungen nicht vorhanden sind.

Anwendbarkeit

Der Zeitaufwand für die Anwendung des Regelwerks ist enorm. Die Anwendung kann jedoch zumindest so gestaltet werden, dass die Belastung der Anforderungssteller gering ist und damit die Akzeptanz gesichert wird.
Die Belastung der Systemanalytiker ist hoch. Sowohl Zeitbedarf, intellektuelle Beanspruchung, wie auch Eintönigkeit der Regelanwendung verursachen bei einer größeren Zahl an Anforderungen Konzentrationsverluste und Demotivation.

5 Bewertung der Ergebnisse

Das nachfolgende Kapitel beschäftigt sich mit der Qualität der Anforderungen, sowie des Anforderungsdokuments, das im Rahmen der Simulation entstanden ist. Dadurch sollen Rückschlüsse auf den Nutzen des Regelwerks ermöglicht werden. Im ersten Teil werden die Anforderungen für sich bewertet und den Ausgangsanforderungen gegenübergestellt. Der zweite Teil behandelt dann das Anforderungsdokument im Gesamten. Zum Schluss erfolgt eine Zusammenfassung der Ergebnisse.

5.1 Qualitätskriterien für einzelne Anforderungen

Schwierige Prüfbarkeit

Bei der Überprüfung der einzelnen Anforderungen fällt zunächst auf, dass in der Literatur zwar eine Vielzahl von Qualitätskriterien beschrieben wird, es jedoch an objektiven Metriken mangelt, mit denen das Erreichen dieser Kriterien gemessen werden kann.

(Rupp07) bildet hier keine Ausnahme. Die dortigen Metriken stellen den prozentualen Anteil von „korrekten" Anforderungen fest, wobei „korrekt" je nach Qualitätskriterium für „eindeutig", „verfolgbar" etc. steht. Die Entscheidungsgrundlage, wann eine Anforderung als „korrekt" eingestuft werden kann, bleibt jedoch unklar oder hält einer kritischen Prüfung nicht stand.[17]

Diese Problematik lässt sich teilweise dadurch erklären, dass einerseits Qualitätskriterien wie „Eindeutigkeit" oder „Verständlichkeit" stark vom Leser der Anforderungen abhängen, andererseits Kriterien wie „Vollständigkeit" bspw. durch den gewünschten Detaillierungsgrad bestimmt sind. Allgemeingültige, projektunabhängige Bemessungsgrundlagen sind so schwer zu schaffen.

Die nachfolgende Bewertung kann diese Probleme nur bedingt lösen. Sie versucht jedoch durch den Vergleich mit den ursprünglichen Anforderungen den Grad der Verbesserung zu verdeutlichen und ferner die Meinung des Anforderungsstellers, sowie unbeteiligter Dritter mit einzubeziehen.

---

[17] Vgl. (Rupp07), S. 358 ff.

5.1.1 Vollständigkeit

Die Prüfung der Vollständigkeit der Anforderungen muss berücksichtigen, dass in der Anforderungserhebung bewusst nur ein Teilaspekt des gewünschten Reisebuchungssystems spezifiziert wurde und für diesen zudem nur die funktionalen Anforderungen. Eine Aussage über die Vollständigkeit der Anforderungen im Gesamten kann damit nicht getroffen werden.

Sind einzelne Anforderungen komplett? Im Vergleich zu den ursprünglichen Anforderungen?

Abhängigkeit vom Detaillierungsgrad

Einzelne Anforderungen sind nach (Rupp07) vollständig, wenn sie „[…] die geforderte und zu liefernde Funktionalität vollständig umschreiben".[18] Dabei muss der gewünschte Detaillierungsgrad mitbetrachtet werden. Ist dieser hoch, müssen Anforderungen mehr Informationen enthalten, um als vollständig zu gelten.

Alleinstehende Betrachtung

Nachdem das Regelwerk solange auf die Anforderungen angewendet wurde bis keine Fehler mehr zu entdecken waren, kann behauptet werden, dass die neuen Anforderungen vollständig gemäß dem SOPHIST-REgelwerk sind. Das bedeutet, dass in jeder Anforderung folgende Informationen enthalten sind:

- Alle beteiligten Objekte (Personen, Systeme etc.) einer Funktionalität samt deren Beschaffenheit, deren Anzahl und deren Rolle (Auslöser, Ausführender)
- Informationen über die Art, wie Funktionalitäten umzusetzen sind
- Der Zeitpunkt und die Häufigkeit, wann Funktionalitäten in Anspruch genommen werden können, sowie evtl. zeitliche Abhängigkeiten
- Evtl. Fehlerfälle und Sonderfälle inkl. aller möglichen Bedingungen
- Evtl. genau spezifizierte Qualitätskriterien

Ergebnisse der Prüfung

Die Überprüfung der Anforderung bestätigt, dass dies nahezu immer gegeben ist. Dort, wo die Prüfung negativ ausfällt, wurde entweder die Anwend-

[18] (Rupp07), S. 28

barkeit einer Regel nicht erkannt oder bewusst auf deren Anwendung verzichtet. Grund hierfür ist, dass ansonsten der gewünschte Detaillierungsgrad überschritten worden wäre bzw. dass das Weglassen von Informationen nach subjektiver Beurteilung zu mehr Verständlichkeit führt.
So wurde wie erwähnt bspw. auf die Angabe der Häufigkeit „immer" verzichtet. Ebenso wurde an einigen Stellen die Information weggelassen, in welcher Bildschirmmaske sich bestimmte Eingabefelder befinden.
Andererseits ist in jeder Anforderung zu erkennen, was das System zur Verfügung stellen muss bzw. was es zu erbringen hat, sowie wer die Funktionen nutzt. Nach Meinung des Projektteams sind auch alle beteiligten Objekte vorhanden und ausreichend spezifiziert. Selbiges gilt für zeitliche Abhängigkeiten und Sonder- bzw. Fehlerfälle. Ebenso ist zu erkennen, wie die Funktionen zu erbringen sind.

Vergleich der Anforderungen

Vergleicht man die Originalanforderungen mit den Ergebnisanforderungen ist unschwer zu erkennen, dass der Grad an Vollständigkeit, wie auch an Detailliertheit deutlich zugenommen hat.
So fehlt in den ursprünglichen Anforderungen in nahezu jedem Fall die Information über den Nutzer und Erbringer von Funktionen, sowie über die beteiligten Systeme. Auch die weiteren, oben erwähnten Informationen sind nicht oder nur sehr grob vorhanden.

Absolute Vollständigkeit

Inwieweit die Anforderungen ausreichend vollständig sind, um für die Implementierung eines realen Systems verwendet werden zu können, kann nur gemutmaßt werden. Die Annahme, dass sie ergänzt um nicht-funktionale Anforderungen (grafische Benutzeroberfläche, technische Randbedingungen etc.) den Großteil der Fragen in den späteren Systementwicklungsphasen beantworten, erscheint jedoch aufgrund ihres Informationsgehaltes sehr plausibel.

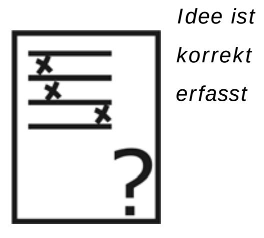
5.1.2 Korrektheit

Entsprechen die Anforderungen dem Wunsch des Anforderungsstellers?

Bei der Bewertung der Anforderungen in Bezug auf ihre Korrektheit wurde besonders auf die Meinung, so wie das fachliche Wissen des Anforderungsstellers geachtet. Es werden zum einen die Angaben aus dem Fragebogen

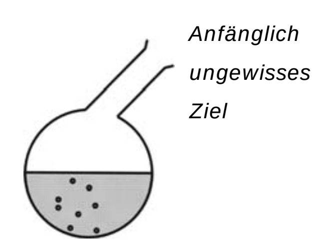
zur Bewertung herangezogen, zum anderen werden die Rückmeldungen des Anforderungsstellers während der Simulation berücksichtigt.

Idee ist korrekt erfasst



Die neuen Anforderungen können als korrekt bezeichnet werden, da der Anforderungssteller nach eigener Aussage die Idee, die hinter der Anforderungserhebung stand in den neuen Anforderungen wieder finden konnte. Er gibt des Weiteren an, dass die neuen Anforderungen die Funktionalitäten zum Buchen einer Reise komplett und ohne inhaltliche oder funktionale Fehler abdecken.

Anfänglich ungewisses Ziel



Der Anforderungssteller merkte allerdings an, dass er nicht ausreichend beurteilen könne, ob die Anforderungen genau das System wiedergeben, von dem er zu Beginn der Erhebung ausgegangen sei. Er begründete dies damit, dass ihm am Anfang noch nicht genau bewusst gewesen wäre, wie das spätere System aussehen solle.

Aufdeckung neuer Sachverhalte

In Bezug auf die neu hinzugekommenen Anforderungen gab der Anforderungssteller an, dass durchaus einige Sachverhalte aufgedeckt werden konnten, die er zu Beginn nicht erwähnt bzw. vergessen oder übersehen hatte. Hier sind als Beispiel die Beziehungen des Reisebuchungssystems zum CRS-Server zu nennen. Ohne die Aufdeckung dieser impliziten Sachverhalte hätten die Anforderungen weder aus Sicht des Projektteams noch aus Sicht des Anforderungsstellers als korrekt bezeichnet werden können.

5.1.3 Konsistenz

Sind die Anforderungen in sich und gegenüber anderen widerspruchsfrei? Im Vergleich zu den ursprünglichen Anforderungen?

Kaum Inkonsistenzen

Überprüft man die neuen Anforderungen, ist zunächst festzustellen, dass keine Widersprüche innerhalb einer bzw. zwischen verschiedenen Anforderungen zu entdecken sind. Es können jedoch an drei Stellen Fehler entdeckt werden, die man als Inkonsistenz bezeichnen kann. So ist in den Anforderungen 31, 33 und 34 (Nummerierung gemäß Kap. 3.6.1) teilweise von „Detailanzeigen für Hotels" und teilweise von „Detailanzeigen für Hotels und Pauschalreisen" die Rede - richtig wäre immer „Hotel". In Anforderung 41 wird nicht angegeben, dass auch die verfügbaren Flüge abzurufen sind - im

Gegensatz zu Anforderung 40, in der dies der Fall ist. Schließlich ist durch Anforderung 64 die Möglichkeit gegeben, dass Reisen in die Reiseliste eingetragen werden, auch wenn sie nicht erfolgreich gebucht wurden. Während beim letzten Fehler evtl. noch Regel 11 („Ermitteln Sie unvollständig spezifizierte Bedingungen") greifen könnte, ist das Projektteam beim Rest der Anforderungen der Meinung, dass hier durch das Regelwerk keine Fehler entdeckt werden können.

Originalanforderungen

In den ursprünglichen Anforderungen sind ebenfalls keine Widersprüche zu entdecken. Als inkonsistent mag allerdings angesehen werden, dass in Anforderung 9 und 22 der Kunde als auswählender Akteur auftritt (eigtl. der Sachbearbeiter) und dass in Anforderung 2 ein nicht gegebener Zeitverlauf dargestellt wird.

Es lässt sich zusammenfassen, dass in den neuen Anforderungen keine grundlegenden Widersprüche oder Inkonsistenzen vorhanden sind. Wie bereits mehrfach angesprochen, kann dies nicht auf das Regelwerk zurückgeführt werden, da dieses nicht daraufhin prüft. Bei umfangreicheren Spezifikationen ist ein schlechteres Ergebnis zu erwarten. Allenfalls wird durch die intensive Beschäftigung mit den Anforderungen durch das Regelwerk auf indirektem Wege die Gefahr von Inkonsistenzen vermindert.

5.1.4 Testbarkeit

Lassen sich Testfälle aus der Anforderung ableiten?

In diesem Abschnitt soll geklärt werden, ob aus den neuen Anforderungen Abnahmekriterien bzw. Testfälle abgeleitet werden können.

Definition Abnahmekriterium

Unter einem Abnahmekriterium versteht Chris Rupp „[...] eine Anweisung für den Test bezüglich einer Anforderung (oder eines Anforderungsteils), welche die Prüfung und Bewertung des erstellten Produktes [...] gegenüber dieser Anforderung (oder des Teils) beschreibt."[19] Abnahmekriterien sind nötig, um später – am Ende der Entwicklung eines Systems – bestimmen zu können, ob wirklich das System entwickelt wurde, das am Anfang in den Anforderungen definiert wurde. In (Rupp07) werden hierzu noch weitere

---

[19] (Rupp07), S. 324.

Arten von Abnahmekriterien beschrieben, auf die nicht näher eingegangen werden soll.[20]

Sie beschreibt außerdem, dass es wichtig ist, die richtigen Abnahmekriterien zu finden. Es werden hier als alternative Verfahren das Black- und das White-Box-Verfahren beschrieben. Beim White-Box-Verfahren ist der Weg entscheidend für die Erfüllung der Anforderungen. Das System muss hier einen genau definierten Weg einhalten, wie die Eingaben in die Ausgabe umgesetzt werden. Beim Black-Box-Verfahren ist nur die Ausgabe wichtig.

Black-Box - White-Box

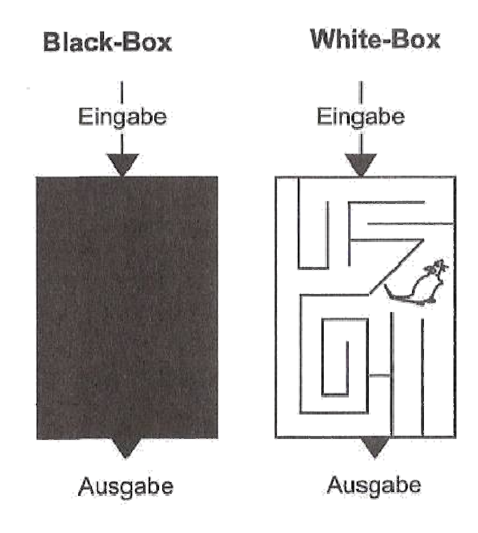
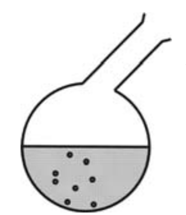


Abbildung 1: Black-Box- und White-Box-Verfahren

(Quelle: (Rupp07), S. 332)

Für den Fall des Reisebuchungssystems sollte das Black-Box-Verfahren angewendet werden. Hierfür spricht zum einen, dass bei diesem System nur das Ergebnis entscheidend ist. Die Reise sollte zuverlässig gebucht werden – alles andere ist sowohl dem Kunden als auch dem Sachbearbeiter im Reisebüro egal. Auch wurden mit dem Anforderungssteller keine Anforderungen bezüglich der technischen Realisierung aufgenommen. So fehlen zum Beispiel sämtliche Anforderungen bezüglich der Kommunikation zwischen dem CRS-Server und dem Reisebuchungssystem.



Black-Box zum Test geeignet

Diesen Ansatz verfolgt auch Bröhl. Er stellt in einem V-Modell dar, für welche Testfälle das Black- bzw. das White-Box-Verfahren geeignet sind. Aus diesem Modell geht hervor, dass für Anwenderforderungen – wie Bröhl die

Black-Box auch nach Bröhl geeignet

[20] Vgl. (Rupp07) S. 327 ff.

Anforderungen auf der ersten Ebene nennt – ausschließlich das Black-Box-Verfahren geeignet ist.[21]

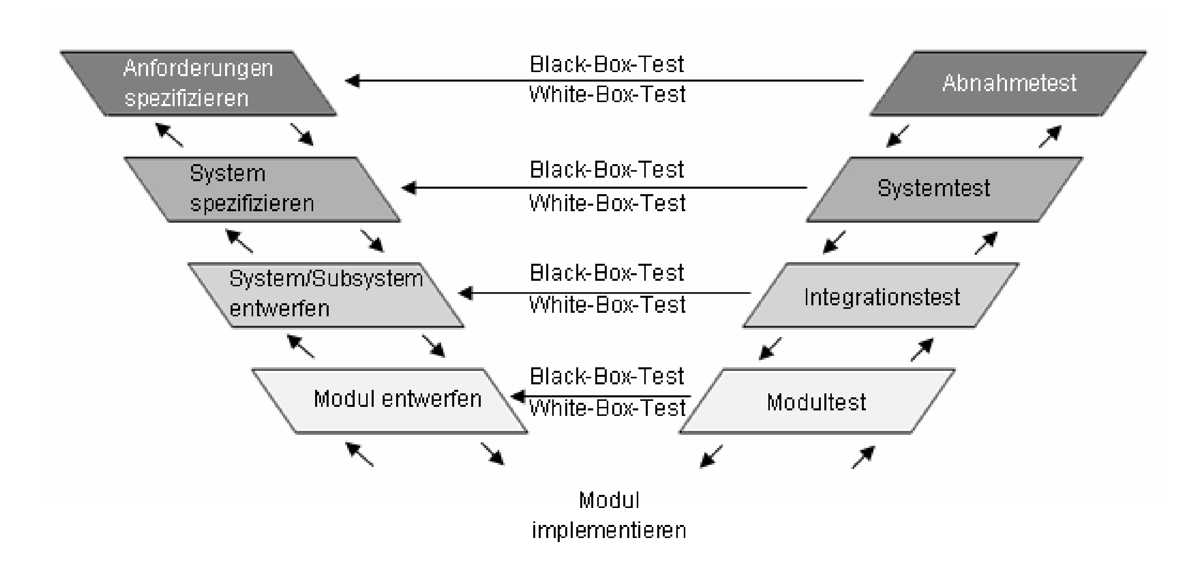


Abbildung 2: V-Modell mit Black-Box- und White-Box-Verfahren

(Quelle: (Bröhl99) nach (Rupp07), S. 333)

Da die Anforderungen für das Reisebuchungssystem ausschließlich Anwenderanforderungen sind, ist hier zum Testen das Black-Box-Verfahren am Besten geeignet.

Verschiedene Testverfahren

Im Rahmen des Black-Box-Verfahrens sollten nach Rupp alle möglichen Kombinationen von Eingaben getestet werden. Da dies aber aus Zeit- und Kostengründen meistens nicht möglich ist, gibt es drei Methoden, nach denen Abnahmekriterien definiert werden können:

- Funktionsabdeckung
- Äquivalenzklassenbildung
- Grenzwertanalyse

Die Anforderungen für das Reisebuchungssystem können nach allen Methoden getestet werden. Um dies ausreichend zu begründen, sollen hier die einzelnen Methoden vorgestellt und jeweils ein Beispiel für das Reisebuchungssystem gegeben werden.

[21] Vgl. (Bröhl99) zitiert nach (Rupp07), S. 333.

Bei der Funktionsabdeckung werden Testfälle gesucht, nach denen die Funktionalität bzw. das Normalverhalten des Systems getestet werden kann.[22] Ein möglicher Testfall nach dieser Methode für das Reisebuchungs-system wäre beispielsweise:

Testfälle nach Black-Box-Verfahren

Anforderung 16:

> *„Das System berechnet die Anzahl der reisenden Kinder aus der Anzahl der nicht mit „-“ belegten Auswahllisten für das Alter der reisenden Kinder.“*

Ereignis:

> *Der Sachbearbeiter im Reisebüro wählt in einer der drei Auswahllisten den Wert „8“ für das Alter des einen mitreisenden Kindes aus.*

Erwartetes Ergebnis:

> *Das Reisebuchungssystem aktualisiert die Anzahl der Reisenden. Es reduziert dabei die Anzahl der reisenden Erwachsenen um 1 und gibt an, dass ein Kind an der Reise teilnimmt.*

Da bei der Funktionsabdeckung lediglich das Normalverhalten abgeprüft werden kann, müssen über die anderen Methoden die Fehlerfälle abgedeckt bzw. abgeprüft werden. Über die Äquivalenzklassenbildung werden Testfälle gleicher Art zusammengefasst. Als „gleichwertig“ können Testfälle dann bezeichnet werden, wenn sie das gleiche Ergebnis zur Folge haben.[23] Als Beispiel für das Reisebuchungssystem wird hier die Frage nach der Reisedauer herangezogen.

Anforderung 8 und 9:

> *„Das System ermöglicht jedem Sachbearbeiter über je genau ein Zahleingabefeld in der Suchmaske die minimale Reisedauer und die maximale Reisedauer der gesuchten Pauschalreisen in Tagen einzugeben.*
>
> *Das System belegt das Zahleingabefeld für die minimale Reisedauer mit 0 und das Zahleingabefeld für die maximale Reisedauer mit 7.“*

---

[22] Vgl. (Rupp07), S. 334
[23] Vgl. (Rupp07), S. 334 f.

Eingabewerte:

Minimale Reisedauer

Maximale Reisedauer

Ausgabewerte:

Reisedauer wie angegeben oder

min. Reisedauer 0 und max. Reisedauer 7

Bedingung:

Die Reisedauer wurde angegeben oder

Die vorbelegte Reisedauer wurde übernommen oder gelöscht

Äquivalenzklassen:

Äquivalenzklasse	Wertebereich
1	Min Reisedauer > 0; Max Reisedauer <> 7
2	Min Reisedauer = 0 oder leer; Max Reisedauer = 7 oder leer

Mögliche Repräsentanten:

Äquivalenzklasse	Wertebereich
1	Min Reisedauer = 7; Max Reisedauer = 14
2	Min Reisedauer leer; Max Reisedauer leer

<u>Abnahmekriterium für die Äquivalenzklasse 1:</u>

Ausgangssituation:

Die Felder für die minimale und die maximale Reisedauer sind mit „0“ und „7“ vorbelegt.

Ereignis:

Der Sachbearbeiter gibt als minimale Reisedauer „7“ und als maximale Reisedauer „14“ ein.

Erwartetes Ergebnis:

Das System gibt nur Reisen aus, die eine Reisedauer zwischen 7 und 14 Tagen haben.

<u>Abnahmekriterium für die Äquivalenzklasse 2:</u>

Ausgangssituation:

Die Felder für die minimale und die maximale Reisedauer sind mit „0“ und „7“ vorbelegt.

Ereignis:

Der Sachbearbeiter löscht die Werte in den Feldern für die minimale und die maximale Reisedauer und vergisst neue Werte einzugeben.

Erwartetes Ergebnis:

Das System gibt nur Reisen aus, die eine Reisedauer zwischen 0 und 7 Tagen haben.

Somit sind auch Testfälle über das Verfahren der Äquivalenzklassenbildung möglich.

Die Grenzwertanalyse baut auf der Äquivalenzklassenbildung auf. Es werden die möglichen Wertebereiche der Äquivalenzklassen betrachtet und als Werte werden dann die Grenzwerte zum Testen verwendet. Dieses Verfahren basiert auf der Erfahrung, dass häufig Fehler gerade in den Grenzwertbereichen auftreten.[24] Das beschriebene Vorgehen ist auch im Fall des Reisebuchungssystems anwendbar. Hierfür kann das obige Beispiel dienen, da dort bereits Grenzwerte verwendet wurden.

Zu beachten ist beim Testen die Erfahrung des Entwicklers. Jeder Entwickler entwickelt mit der Zeit ein Gefühl dafür auf, bei welchen Testfällen gerne Fehler auftreten. Als Beispiele nennt Rupp hier die Eingabe von Nullwerten für Werte, mit denen später weitergerechnet wird, oder die Eingabe von ungültigen Datumswerten oder Sonderzeichen bzw. Umlauten.[25] Auch das Reisebuchungssystem bietet Ansatzpunkte für derartige intuitive Tests:

Anforderung 49:

„Das System ermöglicht jedem Sachbearbeiter den Vor- und Zunamen, die Strasse, die PLZ, den Wohnort und das Geburtsdatum des Kunden in je einem Eingabefeld in der Kundenmaske zu erfassen.“

Testfall:

Es können Umlaute und Sonderzeichen, aber auch überlange Zeichenketten eingegeben werden.

---

[24] Vgl. (Rupp07), S. 336
[25] Vgl. (Rupp07), S. 337 ff.

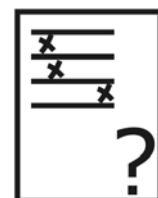
Als Ergebnis der vorhergehenden Ausführungen ist festzuhalten, dass aus den Anforderungen Abnahmekriterien für das Black-Box-Verfahren definierbar sind und die Anforderungen somit testbar sind.
Grund dafür ist die ausreichende Genauigkeit der Anforderungen und der hohe Informationsgehalt.

5.1.5 Eindeutigkeit

Verhindern die Anforderungen Interpretationsspielräume? Im Vergleich zu den ursprünglichen Anforderungen?

Vergleich Ursprungs-anforderungen

Bei einem direkten Vergleich der ursprünglichen mit den bearbeiteten Anforderungen stellt man fest, dass sich der Umfang der einzelnen Anforderungen vervielfacht hat. Rein äußerlich kann dadurch der Eindruck einer Verkomplizierung erweckt werden. Zudem steigt der Zeitaufwand zum Lesen der Anforderungen erheblich.
Tatsächlich jedoch wird der erwünschte Nutzen erreicht. Die Anforderungen haben an Eindeutigkeit gewonnen. Sie bergen für verschiedene Lesegruppen nahezu keine Interpretationsspielräume. Die genauen Bestimmungen und Spezialisierungen, die das Erfüllen der Regeln voraussetzt, mögen zwar an vielen Stellen unnötig oder übertrieben wirken, gewährleisten jedoch, dass alle Zielgruppen unter dem Gelesenen dasselbe verstehen.



Meinung des Anforderungsstellers

In der durchgeführten Simulation wurde der Anforderungssteller mittels eines umfassenden Fragebogens um seine Einschätzung gebeten. Die neuen Anforderungen wurden dabei mit „eindeutig“ beurteil. Es mag erstaunen, dass dem Befragten tatsächlich die Notwendigkeit der Regelwerksanwendung bewusst wurde. Auf die Frage „Glauben Sie, dass die Anforderungserhebung durch das SOPHIST-Regelwerk sicherer geworden ist?“ antwortete der Anforderungssteller mit einem eindeutigen „ja“. Der Grund hierfür liegt in der systematischen Hinterfragung der Anforderungen. Der Anforderungssteller empfand die neuen Anforderungen im Vergleich zu den ursprünglichen Anforderungen tatsächlich als eindeutiger als die ursprünglichen Anforderungen, obwohl diese von ihm stammten. Keine der neuen Anforderungen wurden als unnötig erachtet.

Auch nach Meinung des Projektteams sind die Anforderungen eindeutig. Um die Bewertung jedoch repräsentativer zu gestalten, wurde eine dritte Meinung eingeholt: Eine Umfrage in der ZF Informatik bestätigte die Eindeutigkeit größtenteils. Auch wenn der Umfang für vermeintlich einfache Aussagen zunächst unverhältnismäßig schien, musste eingeräumt werden, dass diese genauen Formulierungen durchaus sinnvoll sind und vor schnell begangenen Interpretationsfehlern schützen. Es wurde bestätigt, dass ungenaue Anforderungen häufig zu Problemen bei der Programmierung führen und einen enormen Kostenfaktor darstellen können. Solche „unsauberen" Anforderungen werden häufig von Entwicklern nicht erkannt, weil ihnen die Aussagen so trivial erscheinen. Mehrdeutigkeiten werden überflogen und automatisch durch Eigeninterpretationen ersetzt, was häufig zu einem Mehraufwand aufgrund späterer Änderungswünsche und damit zu erhöhten Kosten führt. Dies kann mit den Ergebnisanforderungen der Simulation vermieden werden.

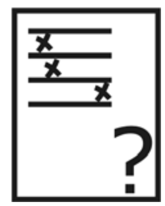
Meinung des Projekt-teams

5.1.6 Verständlichkeit

Sind die Anforderungen verständlich? Im Vergleich zu den ursprünglichen Anforderungen?

Um die Verständlichkeit der Anforderungen zu bewerten wurden zwei verschiedene Betrachtungsweisen gewählt. Einerseits bewertete der Anforderungssteller die Anforderungen, andererseits wurde ein sachverständiger Dritter hinzugezogen. Beide sollten sowohl die neuen Anforderungen für sich, aber auch im Vergleich zu den alten Anforderungen auf ihre Verständlichkeit hin bewerten.

Der Anforderungssteller bewertete die neuen Anforderungen als verständlich. Er gab an, dass er durch die neuen Anforderungen eine klare Vorstellung von dem dargestellten Reisebuchungssystem erhält. Die neuen Anforderungen würden sehr detailliert beschreiben, wie das System die Funktionen und Schritte bei der Buchung einer Reise umsetzt.



Verständ-licher durch Genauigkeit

Er gab an, dass die neuen Anforderungen im Vergleich zu den alten Anforderungen um ein Vielfaches verständlicher wären. Zum einen, weil die Verwirrungen durch falsch oder synonym verwendete Begriffe eliminiert werden konnten (z. B. Reiseart bedeutet nicht dasselbe wie Reise). Zum anderen

weil einzelne Sachverhalte – wie die Eigenschaften einzelner Eingabefelder (z. B. Verpflegungsart, Hotelkategorie,...) – durch die exakten Beschreibungen in den neuen Anforderungen besser und deutlicher dargestellt wurden.

Verständlich für Außenstehende

Die Beurteilung durch einen sachverständigen Dritten bestätigt die Bewertung des Anforderungsstellers. Die neuen Anforderungen wurden sowohl allein stehend als auch im Vergleich zu den alten Anforderungen als verständlicher eingeschätzt. Die Begründung war ebenfalls die höhere Genauigkeit und die ausführlicheren Beschreibungen der einzelnen Sachverhalte.

Komplexität versus Verständlichkeit

Beide Betrachter bemerkten allerdings, dass die Verständlichkeit der neuen Anforderungen zum Teil unter der enormen Komplexität leidet. Dies verdeutlicht gerade Anforderung 23. Hier muss allerdings bemerkt werden, dass der Sachverhalt, der in Anforderung 23 beschrieben wird, an sich besonders komplex ist. Es ist fraglich, ob dieser Sachverhalt ohne Anwendung des SOPHIST-REgelwerks besser herausgearbeitet hätte werden können.

Wichtig ist festzuhalten, dass der Anforderungssteller und der Dritte bestätigten, dass der Zuwachs an Verständlichkeit durch die ausführlichere Darstellung die Abnahme durch die komplexere Darstellung übertrifft.

5.2 Qualitätskriterien für die gesamten Anforderungen

Inwieweit das Anforderungsdokument im Gesamten bewertet werden soll, mag fraglich erscheinen, nachdem das SOPHIST-REgelwerk sich hauptsächlich auf die Anforderungserhebung und nicht auf die -dokumentation bezieht. Das Projektteam ist jedoch der Meinung, dass Erhebung und Dokumentation in starkem Maße zusammenhängen.
Gerade um die Vorteile von natürlichsprachlichen Anforderungen (Verständlichkeit, Detailliertheit) nutzen zu können, müssen diese sinnvoll strukturiert aufbereitet werden. Mit wachsenden Anforderungsdokumenten nimmt die Notwendigkeit zu, da hier durch die Unübersichtlichkeit von natürlichsprachlichen Anforderungen Qualitätskriterien wie Vollständigkeit oder Konsistenz zunehmend gefährdet werden.

5.2.1 Umfang und Struktur

Wie strukturiert ist die Spezifikation im Vergleich zu den ursprünglichen Anforderungen?

Zunächst ist zu erkennen, dass das Regelwerk keine Vorgaben zur Strukturierung oder Gliederung von Anforderungen in Form von Regeln enthält. Damit sind die Ergebnisanforderungen (wie auch die Originalanforderungen) unstrukturiert.

Empfehlungen in (Rupp07)

In (Rupp07) werden jedoch an anderer Stelle grobe Empfehlungen für die Dokumentation natürlichsprachlicher Anforderungen gegeben.[26] In diesen werden mehrere Detaillierungsebenen vorgeschlagen, wobei in den oberen Ebenen Diagramme für eine Übersicht sorgen und in der untersten Ebene Prosaanforderungen zur Detaillierung eingesetzt werden.

Diesem oder einem ähnlichen Vorschlag ist zu folgen, da bereits bei den wenigen Anforderungen aus der Simulation eine gewisse Unübersichtlichkeit zu erkennen ist.

Zusammenhängende Anforderungen

Wünschenswert wäre dennoch eine bessere Integration einer Gliederungssystematik in das Regelwerk. Nach Meinung des Projektteams könnte dies zur Verständlichkeit der Anforderungen beitragen. Es sollten damit verbundene Anforderungen (Anwendungsfälle, zeitl. Abläufe, Fehlerfälle) besser dargestellt werden können, bspw. indem Benennungskonventionen oder Randbedingungen, die sich nur auf diese Anforderungen beziehen, einmalig festgelegt werden. Da eine derartige Regelung fehlt, müssen die Anforderungen so kontextunabhängig wie möglich gestaltet werden, weswegen sehr viele Informationen in diese aufgenommen werden müssen.

Ist der Umfang aller Anforderungen für den Informationsgehalt angemessen?

Um das Kriterium eines angemessenen Umfangs zu erfüllen, sollten keine oder nur wenige redundante Informationen in den Anforderungen enthalten sein. Zudem sollten unwichtige Informationen vermieden werden.

[26] Vgl. (Rupp07), S. 63, S. 211 ff., S. 221 ff.

Am Beispiel der Suchmaske des Reisebürosystems kann dies leicht überprüft werden. Die Anforderungen 3 - 22 beschreiben, wie die Maske aufgebaut ist und bedient wird. Die Anzahl von 19 Anforderungen erscheint hierfür relativ hoch. Beim Durchlesen der Anforderungen können kaum unwichtige Informationen erkannt werden, jedoch viele Wiederholungen (bspw. werden die einzelnen Suchfelder mehrfach erwähnt). Dies ist wiederum aus der Kontextunabhängigkeit begründet. Hier würden die eben vorgeschlagenen fallbezogenen Benennungskonventionen helfen (wie „Suchfelder" statt der Aufzählung der einzelnen Felder). Berücksichtigt man, dass zu den funktionalen Anforderungen im realen Einsatz noch die Spezifikation der Benutzerschnittstelle käme, ergäbe sich eine weitere Erhöhung der Redundanz, die ebenfalls geregelt werden müsste.

Redundanz und Wiederholungen

Zusammenfassend kann gesagt werden, dass durch das Regelwerk keine strukturierten Anforderungen entstehen, was sich sowohl auf Übersichtlichkeit wie Umfang negativ auswirkt. Eine Erweiterung des Regelwerks wäre hier empfehlenswert.

5.2.2 Vollständigkeit

Ist die Anforderungsspezifikation formal vollständig?

Zur Bewertung der Vollständigkeit bezüglicher der Anforderungen als Gesamtheit sollen die Anforderungen unter den Bewertungskriterien für eine Anforderungsspezifikation betrachtet werden.
Als Beispiel einer Anforderungsspezifikation soll hier die Spezifikation nach IEEE Standard 830 verwendet werden, die in der folgenden Tabelle dargestellt wird. Schon aus dieser knappen Darstellung einer Anforderungsspezifikation ist erkennbar, dass das SOPHIST-REgelwerk in diesem Bereich noch einige Mängel aufweist.

Kapitel	Inhalt
1	Einführung
2	Glossary
3	Spezifikation der Kundenanforderungen
4	Systemarchitektur
5	Spezifikation der Systemanforderungen
6	Systemmodelle
7	Evolution des Systems
8	Anhänge
9	Index

Tabelle 4: Anforderungsspezifikation nach IEEE-Standard 830 (Quelle: IEEE Standard 830 (IEEE98), zitiert nach Ebert (Ebert05), S. 115)

Keine Angaben zur Strukturierung

Das SOPHIST-REgelwerk macht keine Angaben dazu, wie die Anforderungen geordnet bzw. zusammengefasst werden sollten. In (Rupp07) finden sich keinerlei Hinweise oder Angaben, wie die Kapitelstruktur einer Anforderungsspezifikation aussehen sollte. Der Anwender erhält keinerlei Hilfestellung, wie die Anforderungen gruppiert werden sollten.

Keine Angaben zur Nummerierung

Des Weiteren liefert (Rupp07) keine Hinweise bezüglich der Nummerierung von Anforderungen. Hier stellt sich die Frage, ob sich neu ergebende Anforderungen generell an das Ende der Liste der Anforderungen zu stellen sind, oder ob sie an der Stelle, an der sie sich ergeben, eingefügt werden sollten. Es ist außerdem unklar, wie bei einer Verfeinerung von Anforderungen zu verfahren ist. Soll hier eine Untergliederung im Sinne von Unterkapiteln vorgenommen werden (z. B. 1.1..., 1.2...) oder soll für jede „Unteranforderung" eine eigenständige Anforderung erstellt werden?
Das Projektteam einigte sich hier darauf während der Simulation die neuen Anforderungen mit der Nummer „Anzahl Anforderungen + 1" an der Stelle ihrer Entstehung in die Liste der Anforderungen aufzunehmen. „Unteranforderungen" wurden während der Simulation innerhalb der Ursprungsanforderung weitergeführt. Es sollte so ein besserer Überblick gewährleistet werden. Für die endgültige Aufstellung der Anforderungen wurden dann alle Anforderungen fortlaufend auf einer Ebene durchnummeriert aufgelistet.

Glossar hilfreich

Hilfreich für die formale Vollständigkeit ist allerdings das Glossar. Dessen Erstellung wird in (Rupp07) vorgeschlagen. Hierdurch werden die eigentlichen Anforderungen übersichtlicher und wichtige Begriffe sind zentral an einer Stelle gesammelt.

Im Vergleich zu den ursprünglichen Anforderungen?

Ergebnis

Im Vergleich zu den ursprünglichen Anforderungen ist die formale Vollständigkeit allein durch die Verwendung eines Glossars enorm gestiegen. Der Überblick über die im System verwendeten Begrifflichkeiten kann so besser gewährt werden und es können so Missverständnisse und Konflikte wegen Begriffsdefinitionen weitgehend vermieden werden.

5.2.3 Weiterverwendbarkeit

Können die Anforderungen weiterverwendet werden (Analysemodell, Vertrag)? Im Vergleich zu den ursprünglichen Anforderungen?

Weiteverwendung gewünscht

Auch wenn in diesem konkreten Beispiel keine Weiterverwendung des Anforderungskatalogs stattfindet, muss dieser dennoch darauf überprüft werden. Schließlich ist im praktischen Fall die Weiterverwendung der Anforderungen in jedem Fall das Ziel. Ohne diese ist die Anwendung des Regelwerks und die Erstellung eines solchen Katalogs nutzlos. Schließlich sind die gesammelten Anforderungen der Grundstock für Analysemodelle und/oder ein Lasten- und Pflichtenheft, welches wiederum Bestandteil von Aufwandsschätzungen und Verträgen ist.

Untauglichkeit der Ursprungsanforderungen

Die ursprünglichen Anforderungen wären ohne eine weitere Bearbeitung nicht für ein Modell oder als Grundlage für einen Vertrag brauchbar gewesen. Sie waren mehrdeutig und gaben die eigentlichen Erwartungen an das zu erstellende System nicht verbindlich wieder. Es war nicht klar, welche Leistungen das System ermöglichen sollte oder wie es diese zur Verfügung stellt. Somit konnten kaum Rechte auf bestimmte Leistungen verständlich herausgearbeitet, geschweige denn wirksam rechtlich geltend gemacht werden.

Festlegung des genauen Leistungsumfangs möglich

Durch die Anwendung des SOPHIST-Regelwerks legt der Anforderungskatalog den Umfang des Programms genau fest. Jedoch stellt dieser keine direkte Vorlage zur Programmierung dar, die ohne große Überlegungen übernommen werden kann. Er führt aber weg von allgemeinen, missverständlichen Anforderungen.

Damit der Katalog sämtliche Aspekte der neuen Anwendung betrachtet, müssen jedoch Bestimmungen und Wünsche über das Graphical User Interface (GUI) und weitere nicht-funktionale Anforderungen beigefügt werden. Da jedoch durch das Regelwerk nur funktionale Anforderungen erkannt und betrachtet werden können, müssen diese Dokumente gesondert erhoben und erfasst werden.

Weitere Dokumente nötig

5.3 Anzahl der Anforderungen

Wie entwickelte sich die Anzahl der Anforderungen?

Die Anzahl der Anforderungen hat sich von 19 plus 5 Glossareinträge auf 68 Anforderungen plus 22 Glossareinträge erweitert. Die neuen Anforderungen lassen sich in folgende Kategorien einteilen:

- ◇ Direkter Bezug zu ursprünglichen Anforderungen erkenntlich:
  Detaillierung der Funktionalitäten
  45 Stück
- ◇ Kein direkter Bezug zu ursprünglichen Anforderungen erkenntlich:
  Implizite Anforderungen, neue Anforderungen und Fehler- bzw. Sonderfälle (da ursprünglich keine vorhanden waren)
  23 Stück

Es zeigt sich, dass der größere Teil der neuen Anforderungen zur genaueren Spezifikation gegebener Anforderungen verwendet wird.

5.4 Beispielhafter Vergleich mit der Anforderungsschablone

Inwieweit entsprechen die neuen Anforderungen, denen der SOPHIST-Anforderungsschablone (Satzbaupläne)?

Da Anforderungen, die mit dem SOPHIST-REgelwerk erhoben und formuliert wurden nach Meinung Rupps immer noch nicht optimal sind, schlägt sie die Verwendung von so genannten Anforderungsschablonen vor. Hierdurch könnten Anforderungen syntaktisch korrekt und den Qualitätskriterien für Anforderungen entsprechend formuliert werden.[27]

---

[27] Vgl. (Rupp07), S. 227 ff.

Um zu beurteilen, ob die Anforderungen durch die Anwendung der Anforderungsschablone tatsächlich besser werden, soll hier beispielhaft dieses alternative Vorgehen getestet werden. (Zur näheren Betrachtung der theoretischen Hintergründe zur Anforderungsschablone wird auf (Rupp07) verwiesen werden).

Formulierung eines eigenen Beispiels nach der Anforderungsschablone

Der Sachverhalt, der hier als Beispiel für die Anwendung der Anforderungsschablone verwendet werden soll, ist die Angabe einer Unter- bzw. einer Obergrenze für den Preis der gesuchten Reise.
Als ersten Schritt sieht die Anforderungsschablone vor, dass der zugrunde liegende Prozess definiert wird. Im vorliegenden Beispiel kann der zugrunde liegende Prozess folgendermaßen beschrieben werden:

> *Die Untergrenze bzw. die Obergrenze für den Preis aller zu suchender Pauschalreisen werden eingegeben.*

Hier wurde das Prozesswort „eingeben" nach der Prozesswort-Liste in (Rupp07) gewählt. Die Definition von „eingeben" lautet wie folgt: „Ausschließlich der NUTZER gibt neue Daten ein oder überschreibt vorhandene Daten". Bei diesem Prozesswort wird von einer unselbstständigen Systemaktivität ausgegangen, die eine Interaktion des Benutzers vorsieht.[28] Rupp weist darauf hin, dass die Wahl des Prozesswortes entscheidend ist für das weitere Vorgehen, da es die Semantik der Anforderung maßgeblich beeinflusst.[29]
Da es sich bei dem dargestellten Prozess nach Rupp um eine Benutzerinteraktion handelt, wird nun die Schablone vom Typ 2 angewendet:
DAS SYSTEM...<wem?> DIE MÖGLICHKEIT BIETEN <Prozesswort>.

> → *Das System ... dem Sachbearbeiter im Reisebüro die Möglichkeit bieten etwas einzugeben.*

Die noch bestehende Lücke in der Anforderung wird im nächsten Schritt geschlossen. Es ist nun noch die rechtliche Verbindlichkeit festzulegen. Da es sich bei der Einschränkung des Preises nicht um eine rechtlich bindende Anforderung handelt, aber auch nicht um eine zukünftige, wird hier als Hilfsverb „soll" gewählt. So wird ausgedrückt, dass die Anforderung dringend empfohlen wird.

---

[28] Vgl. (Rupp07), S. 240
[29] Vgl. (Rupp07), S. 228 f.

> → *Das System soll dem Sachbearbeiter im Reisebüro die Möglichkeit bieten etwas einzugeben.*

Es ist erkennbar, dass die Anforderung in ihrem bisherigen Zustand nur als Grundlage für die endgültige Anforderung dienen kann. Es fehlen noch sämtliche Angaben über Objekte etc. Diese sollen im 4. Schritt ergänzt werden.

> → *Das System soll dem Sachbearbeiter im Reisebüro die Möglichkeit bieten die Untergrenze bzw. die Obergrenze für den Preis aller zu suchender Pauschalreisen in je einem Zahleingabefeld einzugeben.*

Natürlich können dieser Anforderung auch noch logische oder zeitliche Bedingungen hinzugefügt werden.

> → *Falls der Kunde den Preis der gesuchten Reise einschränken möchte, soll das System dem Sachbearbeiter im Reisebüro die Möglichkeit bieten die Untergrenze bzw. die Obergrenze für den Preis aller zu suchender Pauschalreisen in je einem Zahleingabefeld einzugeben.*

Da es auch an dieser Stelle noch Fehler in der Anforderung geben kann, empfiehlt Rupp im 6. und letzten Schritt das SOPHIST-REgelwerk anzuwenden. Im vorliegenden Beispiel konnten allerdings keine Fehler mehr über das Regelwerk aufgedeckt werden.

Als Vergleich soll hier nun die Anforderung vorgestellt werden, die sich durch die reine Anwendung des SOPHIST-REgelwerks auf die Originalanforderung ergeben hat:

> *Das System ermöglicht jedem Sachbearbeiter über zwei Zahleingabefelder in der Suchmaske optional die Untergrenze und optional die Obergrenze für den Preis aller zu suchender Pauschalreisen anzugeben.*

Im Vergleich der beiden Anforderungen ist erkennbar, dass die Anforderung nach der Anforderungsschablone die Optionalität durch die Bedingung „Falls der Kunde …" abdeckt, wo hingegen in der Anforderung nach dem SOPHIST-REgelwerk der Zusatz „optional" eingefügt wurde. Zudem ist die rechtliche Verbindlichkeit über das Signalwort „soll" geregelt.

Die Anwendung der Anforderungsschablone erscheint geeigneter als die allein stehende Anwendung des SOPHIST-REgelwerks, da die Anforderungen

so besser und einheitlicher strukturiert sind. Durch die Satzbaupläne ist genau festgelegt, was Hilfsverben in einer Anforderung aussagen (z.B. „soll" bedeutet „dringend empfohlen"). Auch ist die Verwendung der Prozesswortliste in (Rupp07) äußerst hilfreich, da so gleich die Art der Anforderung bzw. der Systemfunktionalität festgelegt werden kann. Die Verwendung der Anforderungsschablone stellt somit eine sicherere Methode zur Erfassung qualitativ hochwertigen Anforderungen dar.

Praktischer Einsatz fraglich

Inwieweit die Anforderungsschablone allerdings für den praktischen Einsatz geeignet ist, ist fraglich. Wenn der Anforderungssteller selbst die Anforderungen mithilfe der Schablone formuliert, dürfte der Aufwand im Gesamten geringer ausfallen, ihn jedoch mehr belasten. Er muss zudem das strukturierte Formulieren von Anforderungen erlernen und gleichzeitig ausreichend kompetent im Fachgebiet sein.
Sind diese Voraussetzungen nicht gegeben oder wird analog zum Vorgehen in der Simulation die Formulierung vom Systemanalytiker vorgenommen, dürfte der Aufwand tendenziell noch steigen. Demgegenüber steht kaum zusätzlicher Qualitätsgewinn.

5.5 Zusammenfassung

Die Bewertung der Ergebnisanforderung zeigt, dass die Ergebnisanforderungen die Qualitätskriterien für einzelne Anforderungen im Wesentlichen erfüllen und den Erwartungen des Anforderungsstellers, sowie des Projektteams gerecht werden.
Durch ihren hohen Informationsgehalt sind sie vollständig und korrekt und ermöglichen das Ableiten von Testfällen, sowie die Weiterverwendung als Vertragsgrundlage, für Aufwandsschätzungen oder Analysemodelle. Dennoch bleiben sie weiterhin verständlich und eindeutig, wenn sie auch teilweise etwas umständlich formuliert sind.
Im Vergleich zu den ursprünglichen Anforderungen ist durchweg eine Verbesserung zu erkennen.
Defizite treten in der Strukturierung der Anforderungen auf, da sich das Regelwerk hauptsächlich auf *einzelne* Anforderungen bezieht und nicht das Gesamtdokument betrachtet. Hier wäre eine bessere Integration in die weiteren Bestandteile eines Anforderungsdokuments und Strategien für übergreifende Anforderungen nötig.

6 Bewertung der Relevanz für die Praxis

Bis zu diesem Punkt konnte festgestellt werden, dass mit dem SOPHIST-REgelwerk qualitativ hochwertige Anforderungen formuliert werden können. Der Weg dorthin wurde überwiegend positiv bewertet, wobei durchaus Kritikpunkte wie der große Aufwand oder die mangelhafte Darstellung des Regelwerks bspw. in (Rupp07) aufgeführt wurden. Aufbauend auf diesen Ergebnissen soll nun untersucht werden, inwieweit das Regelwerk als praxistauglich eingestuft werden kann. Es wird dabei wiederum die Erlernbarkeit, der Aufwand und die Akzeptanz untersucht - dieses Mal jedoch nicht aus Sicht des Projektteams oder des Anforderungsstellers, sondern aus Unternehmenssicht unter Berücksichtigung wirtschaftlicher Erwägungen. Ferner wird an dieser Stelle auch untersucht, wie verbreitet das Regelwerk in der Praxis ist und welche Unterstützung es für dessen Anwendung gibt.

6.1 Erlernbarkeit des Regelwerks in der Praxis

Gibt es Schulungen durch die SOPHIST Group bezogen auf das SOPHIST-Regelwerk und dessen Anwendbarkeit?

Da in der Praxis Mitarbeiter ausreichend trainiert und geschult werden müssen, um ihre Arbeit erfolgreich durchführen zu können, ist es für die Bewertung der Praxisrelevanz von Bedeutung, ob es Schulungen zum SOPHIST-REgelwerk gibt und wie diese aussehen.

Schulungen bezüglich RE und RM enthalten auch SOPHIST-REgelwerk

Schulungsveranstaltungen, in denen *ausschließlich* das SOPHIST-REgelwerk geschult und trainiert wird, gibt es nach Angaben der SOPHIST-Group nicht. Das Regelwerk wird lediglich als Teil von Schulungen zum Thema „Requirements Engineering" angeboten.

Diese Schulungen sind auf zwei Tage ausgelegt und kosten 1.020 Euro. Die Inhalte der Schulungen sind zum einen die Grundlagen des Requirements Engineering und des Requirements Management, es werden aber auch Schulungen angeboten, in denen der Fokus auf der Erhebung und Dokumentierung von Anforderungen in Prosa liegt.[30] Das Regelwerk hat nach Angaben der SOPHIST-Group dort nur eine untergeordnete Bedeutung.

[30] (SOPHIST07)

Aus dieser Tatsache lassen sich eine geringe Praxisrelevanz und wenig Interesse am Regelwerk aus der Wirtschaft ableiten.

Haben Mitarbeiter in der Praxis überhaupt die nötige Zeit, um sich in das Regelwerk einzuarbeiten?

Im Zusammenhang mit der Praxisrelevanz ist es nicht nur interessant, ob es Schulungen für das SOPHIST-REgelwerk gibt, sondern es ist auch wichtig zu wissen, ob Mitarbeiter in der Praxis überhaupt die nötige Zeit aufbringen können, um sich in das Regelwerk einzuarbeiten.

Als Bewertung soll hier zum einen die Einschätzungen des Projektteams herangezogen werden, zum anderen wurden Mitarbeiter aus der SAP-Betreuung der Boehringer Ingelheim Pharma GmbH & Co. KG zu ihrer Meinung bezüglich des SOPHIST-REgelwerks befragt.

Einschätzung des Projektteams

Nach Einschätzung des Projektteams ist der Zeitaufwand für die Lektüre sowie für Schulungen durchaus akzeptabel. Mit zwei Tagen überschreiten die Schulungen nicht das übliche Maß und sind damit auch von der finanziellen Perspektive aus durchführbar. Dies gilt umso mehr dadurch, dass in den Schulungen größtenteils weitere (relevante) Themen aus dem Requirments Engineering angesprochen werden. Das Projektteam geht somit davon aus, dass der Zeitaufwand für das Erlernen des Regelwerks von Unternehmen akzeptiert wird.

Einschätzung von Entwicklern

Dieser Eindruck des Projektteam konnte auch durch eine Befragung von Mitarbeitern aus der SAP-Betreuung der Boehringer Ingelheim Pharma GmbH & Co. KG bestätigt werden. Den Mitarbeitern, die in der Betreuung der Anwender, sowie der Anpassung der verschiedenen SAP-Module tätig sind, wurde das SOPHIST-REgelwerk vorgestellt. Sie erhielten einen Überblick über die verschiedenen Regeln sowie deren Anwendung und Nutzen. Als Leitlinie für den nötigen Zeitaufwand für die Einarbeitung wurden die Erfahrungen des Projektteam vorgestellt.

Den Mitarbeitern erschien der durch das Projektteam ermittelt Aufwand für die Erlernung als gerechtfertigt und auch im Bezug auf das Ergebnis akzeptabel. Die einstimmige Meinung war, dass sowohl der Zeitaufwand für die Lektüre als auch für die Einarbeitung mit Hilfe von Beispielanforderungen

von den jeweiligen Führungspersonen auf jeden Fall akzeptiert werden würde.

6.2 Ressourcenverbrauch

6.2.1 Zeitaufwand

Steht der Zeitaufwand in Relation zum erzielten Ergebnis?

Vergleich der Original- mit den Ergebnisanforderungen

Zur Beantwortung sollen zunächst die Ergebnisse aus der Bewertung des Vorgehens und der Ergebnisse in Erinnerung gerufen werden. Dort wurde der Zeitaufwand mit ca. 67 Minuten pro Anforderung als enorm hoch eingestuft. Ebenso hieß es, dass dieser im realen Einsatz wegen des wohl geringeren Detaillierungsgrades etwas niedriger sein dürfte. Die Qualität der Ergebnisanforderungen wurde dagegen als sehr zufriedenstellend erachtet.

Nur diese zwei Aspekte betrachtend, erscheint der Aufwand für die Anforderungen als akzeptabel. Das Projektteam kennt keine Methode, um im Nachhinein Anforderungen schneller auf dieses Qualitätsniveau zu bringen. Im Kontext einer realen Projektumgebung dürfte dieses Qualitätsniveau unter den meisten Rahmenbedingungen jedoch nicht notwendig sein.
Ausnahmen bilden hier sicherheitskritische Anwendungen. Für einfache Geschäftsanwendungen wird der Zugewinn an Prozesssicherheit den Aufwand nicht übersteigen, auch wenn man berücksichtigt, dass Fehler in der Systemanalyse Kosten verursachen, die um ein Vielfaches höher sind als bei Fehlern in späteren Entwicklungsphasen.

Inwieweit ist der Zeitaufwand abhängig vom Umfang der Anforderungen?

In der Simulation zeigte sich, dass mit der Analyse von nur wenigen Anforderungen bereits ein starker Konzentrationsverlust einhergeht, der durch das monotone, aber in der Gesamtheit intellektuell anspruchsvolle Anwenden der Regeln zu erklären ist. Als Größenordnung für das merkliche Nachlassen der Konzentration können hier fünf bis zehn Anforderungen bestimmt werden. Die dadurch notwendigen Pausen verlängern die Anforderungsanalyse weiter.

Die Anwendbarkeit des Regelwerks auf ein großes System mit mehreren hundert Anforderungen in der Praxis erscheint damit auch von dieser Betrachtungsweise her fraglich. Sofern es sich um kein sicherheitskritisches System handelt, steht nicht nur die finanzielle Belastung dagegen, sondern auch der kumulierte Zeitbedarf, der die Systemanalyse erheblich verlängert. Letzterer kann auch nur bedingt durch parallel arbeitende Systemanalytiker verringert werden, da ein Überblick über die Anforderungen für das Erkennen von impliziten Anforderungen unumgänglich ist.

Eignung für kleinere Projekte

Ausgehend von dieser Bewertung ist das Projektteam der Meinung, dass das SOPHIST-REgelwerk nur für kleinere bis mittlere Projekte geeignet ist.

Inwieweit ist der Zeitaufwand abhängig von der Qualität der Anforderungen?

Die Qualität der Anforderungen hat starken Einfluss auf die Zeitdauer der Anwendung. Zwar benötigt die Anwendung der 25 Regeln einen festen Grundzeitbedarf, jedoch ist bei vielen entdeckten Fehlern die Notwendigkeit für die Dokumentation von Rückfragen und vor allem für das regelkonforme Umformulieren der Anforderungen hoch. Wie in der Simulation zu erkennen war, ist deren Zeitbedarf nicht zu vernachlässigen.

Auch ist bei lückenhaften Anforderungstexten der Zeitaufwand höher, da hier von einem Ansteigen der Iterationszahl auszugehen ist. Werden viele wesentliche Anforderungen vergessen, kann davon ausgegangen werden, dass im Gegensatz zur Simulation nicht der Großteil im ersten Interview mit dem Anforderungssteller erkannt werden kann und später entdeckte Anforderungen größere Korrekturen auslösen.

Implizite Anforder-ungen

Für die Praxis bietet es sich daher wie erwähnt an, qualitativ hochwertige Anforderungen als Basis zu nehmen. Diese sollten zumindest schriftlich vom Anforderungssteller fixiert sein und optimalerweise von diesem selbst auf Qualität vorgeprüft werden.

Hochwertige Basis

Ist dies nicht möglich, da der Anforderungssteller fachlich oder zeitlich nicht dazu in der Lage ist, wird das Anwenden des Regelwerks innerhalb eines vernünftigen Aufwands schwierig.

Aus den drei eben bewerteten Punkten zeigt sich, dass das Regelwerk nur unter bestimmten Vorraussetzungen wirtschaftlich angewendet werden kann. So sollte mit dem Scheitern des Entwicklungsprojekts ein großes Schadenspotential verbunden sein, ferner sollte das Projekt einen kleinen bis mittleren Umfang besitzen und schließlich muss eine ausreichende Qualität der Ausgangsanforderungen vorhanden sein.

Voraussetzungen Einsatz

Als weitere Möglichkeit bietet es sich an das Detaillierungsniveau niedrig anzusetzen und evtl. nur die Grundgedanken des Regelwerks in der Prüfung anzuwenden. Nur so kann vom hohen ermittelten Aufwand abgekommen werden, der für Standardanwendungen nicht akzeptabel ist. Die genauen Auswirkungen bei einer weniger genauen Anwendung des Regelwerks auf den Zeitaufwand, aber auch auf die Qualität lassen sich jedoch nur über weitere Versuche bestimmen.

6.2.2 Kosten

Sind Werkzeuge o. ä. nötig, um das Regelwerk sinnvoll einzusetzen?

Die Kosten für den Einsatz des Regelwerks sind zunächst nur die Gehaltskosten, die während der Schulung und während der Anwendung anfallen. Zwingende zusätzliche Kosten sind nicht gegeben, da für die Anwendung des Regelwerks kein Werkzeug erforderlich ist.

Leider können die Anwender des Regelwerks auch nicht auf ein solches Werkzeug zurückgreifen, falls sie es wünschen, da nach Angaben der SOPHIST-Group und eigenen Recherchen kein kommerzielles Werkzeug dafür verfügbar ist. Der Zeitaufwand und damit die Gehaltskosten lassen sich so nur durch selbst entwickelte Vorlagen verringern. Für diese gilt, dass sie in vernachlässigbarer Zeit mit gängiger Bürosoftware zu erstellen sind, jedoch auch ein vergleichsweise geringes Sparpotential aufweisen.

Weitere Kosten entstehen, wenn die Anwender an einer Schulung zum Regelwerk teilnehmen. Die Kosten hierfür sind in Kap. 6.1 aufgeführt. Dabei muss beachtet werden, dass in den Schulungen nicht nur das Regelwerk, sondern auch weitere Techniken des Requirements Engineering behandelt werden. Die Schulungsaufwendungen lassen sich damit nur anteilig veranschlagen.

Schulungskosten

6.3 Akzeptanz des Regelwerks

6.3.1 Akzeptanz von der Entwickler-Seite

Was halten Ermittler/Entwickler in der Praxis vom SOPHIST-REgelwerk?

Um beurteilen zu können, was Entwickler aus der Praxis vom SOPHIST-REgelwerk halten, wurden in der Boehringer Ingelheim Pharma GmbH & Co. KG verschiedene Mitarbeiter aus dem IT-Bereich befragt. Allen Befragten wurde zunächst das Regelwerk vorgestellt und anhand der Anwendung auf Beispielanforderungen der Nutzen des SOPHIST-REgelwerks aufgezeigt.

Inhalt akzeptabel - Formulierung zu hochtrabend

Die Entwickler der Boehringer Ingelheim Pharma GmbH & Co. KG waren der Ansicht, dass das SOPHIST-REgelwerk vom Inhalt und dem erzielten Ergebnis her akzeptabel sei. Sie bemängelten allerdings, dass die Formulierung der einzelnen Regeln zu „hochtrabend" wäre. Es wurde kritisiert, dass die Regeln zum Großteil „so kompliziert" beschrieben seien, dass sie nicht ohne Weiteres zu verstehen wären. Diese Feststellung deckt sich auch mit den Erfahrungen des Projektteams (siehe Kapitel 4.1.2). Die Befragten stellten – wie auch die Teammitglieder – fest, dass die grammatikalischen Fachbegriffe, die in den Regeln verwendet werden, nicht ohne weitere Nachforschungen verständlich sind. Die Voraussetzung eines umfassenden – zum Teil auch spezielleren – grammatikalischen Wissens erschien den Entwicklern als übertrieben. Ihrer Ansicht nach kann ein derart detailliertes Wissen nicht vorausgesetzt werden.

Anwendung abhängig von Größe der Entwicklung

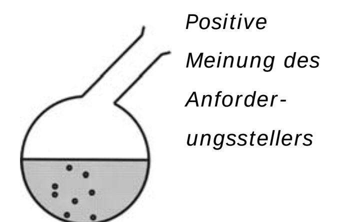
Bei der Frage, ab wann das SOPHIST-REgelwerk als einsetzbar erachtet wird, nannten die Befragten verschiedenen Kriterien. Zum einen sei die Größe des zu entwickelnden Programms sehr entscheidend. Für die Erstellung einer kurzen Auswertung (ca. 1000 – 2000 Zeilen Quellcode, 3 – 5 Bildschirmbilder) würde das SOPHIST-REgelwerk sicher nicht eingesetzt werden, da der Aufwand für die Bearbeitung der Anforderungen viel zu groß ist im Vergleich zur Entwicklung und zu eventuell nötigen Anpassung aufgrund von Fehlinterpretationen oder Missverständnissen. Ab einer Entwicklungsdauer von mehr als 20 Manntagen scheint den Befragten die Anwendung des SOPHIST-REgelwerks sinnvoll. Großen Nutzen versprechen sich

die Entwickler beim Einsatz des Regelwerks bei einer Entwicklung, bei der mehrere Anwender beteiligt sind. Hier könnten dann leichter Unstimmigkeiten zwischen den Anforderungen der verschiedenen Anwender ermittelt und beseitigt werden.

6.3.2 Akzeptanz von der Anwender-Seite

Was halten Anwender in der Praxis vom SOPHIST-REgelwerk?

Hier wird zum einen auf die Ansichten des Anforderungsstellers aus der Simulation zurückgegriffen, zum anderen wird aufgezeigt, wie Entwickler die Meinung des Fachbereichs einschätzen.



Positive Meinung des Anforder-ungsstellers

Der Anforderungssteller aus der Simulation empfand die Anwendung des Regelwerks und auch die damit erzielten Ergebnisse als sehr gut. Den zeitlichen Aufwand für die Anwendung empfand er als durchaus gerechtfertigt und akzeptabel. Auch hatte er das Gefühl, dass er durch das Regelwerk von den Anforderungsermittlern besser verstanden wurde. Seiner Meinung nach werden den Anforderungen durch das SOPHIST-REgelwerk systematischer erhoben und bearbeitet. Sehr hilfreich empfand er, dass durch das Regelwerk Schnittstellen und Begriffsverwirrungen aufgedeckt werden, die auch zum Großteil durch seine eigenen Beschreibungen verursacht wurden.
Der Anforderungssteller geht auch davon aus, dass er – sollte er Anforderungen erheben müssen - das SOPHIST-REgelwerk selbst anwenden wird. Er ist somit überzeugt, dass das SOPHIST-REgelwerk durchaus für die Praxis relevant ist.

Keine Akzeptanz in Fachab-teilung

Die Entwickler der Boehringer Ingelheim Pharma GmbH & Co. KG sind eher skeptisch was die Akzeptanz des SOPHIST-REgelwerks in der Fachabteilung betrifft. Sie gehen davon aus, dass die Anwender aufgrund der komplizierten Formulierung der Regeln und auch wegen dem Umfang des Regelwerks dem Einsatz des SOPHIST-REgelwerks eher mit Zurückhaltung und Ablehnung gegenüber stehen. Ihrer Meinung nach werden Mitarbeiter aus zum Beispiel der Produktion kein Verständnis aufbringen, wenn sie beispielsweise mit unvollständig spezifizierten Prozesswörtern und ähnlichen Fachbegriffen konfrontiert werden.

Das SOPHIST-REgelwerk sollte nach Meinung der Entwickler zum einen vom Umfang her auf 10 prägnante, kurz und verständlich formulierte Regeln reduziert werden, zum anderen sollte nicht so viel grammatikalisches Wissen vorausgesetzt werden. So könnte die Akzeptanz in den Fachbereichen erlangt werden. In der Form wie das SOPHIST-REgelwerk in (Rupp07) vorgestellt wird, wird kein Fachbereich das Regelwerk oder dessen Anwendung akzeptieren.

6.4 Verbreitungsgrad des Regelwerks

6.4.1 Etablierung in der Wirtschaft

Gibt es Unternehmen, die das SOPHIST-REgelwerk in ihren Projekten eingesetzt haben?

Wichtig im Bezug auf die Praxisrelevanz ist die Etablierung der jeweiligen Methode. Wird eine Methode von bekannten Unternehmen in großen Projekten eingesetzt, so kann man davon ausgehen, dass die Methode auch gut, erprobt und sinnvoll ist.

Hier soll nun eine Auflistung von Firmen und ihren Projekten folgen, die – nach Aussage der SOPHIST-Group – das SOPHIST-REgelwerk auch tatsächlich eingesetzt haben.

Unternehmen	Einsatzbereich / Projekt
Deutsche Flugsicherung	Tower-, Center-, Wetter- und Kommunikationssysteme
Grundig Intermedia System	Architektur eines Infotainmentsystems
Skyguide	ATMAS,...
Daimler Chrysler	Telematik, Infotainmentsysteme, Forschung,...
Deutsche Bahn AG	Magnetschwebebahn
Siemens AG	Medizintechnik, Dematic,...
Deutsche Post	Logistiksystem
EADS, ZF Lenksysteme	Waffensysteme
Boehringer Ingelheim	Suchmaske für ein Dokumentenmanagementsystem
...	...

Tabelle 5: Einsatzbereiche des SOPHIST-REgelwerks
(Quelle: http://2005.reconf.de/Sophist_Rupp.pdf, abgerufen am 9.2.07)

Wie aus der Tabelle ersichtlich wird, hat unter anderem die Firma Boehringer Ingelheim mit der SOPHIST-Group zusammengearbeitet - genauer der Medizin-Bereich von Boehringer Ingelheim. Die Zusammenarbeit bestand in der Erstellung einer Suchmaske für ein Dokumentenmanagementsystem. Die Mitarbeiter der SOPHIST-Group nahmen dabei die Anforderungen der Anwender von Boehringer Ingelheim auf, schrieben die User Requirements und erstellten die Spezifikation auf Basis derer dann die Suchmaske entwickelt wurde.

Zusammenarbeit SOPHIST-Group und Boehringer Ingelheim

Des Weiteren führte die SOPHIST-Group bei Boehringer Ingelheim mehrere Workshops und Seminare zum Thema „Requirements Engineering" durch. Nach Auskunft der Mitarbeiter von Boehringer Ingelheim wird zwar das komplette SOPHIST-REgelwerk nicht praktisch angewendet; es wurden lediglich einige der grundlegenden Prinzipien in die tägliche Arbeit übernommen.

6.4.2 Andere Studien und Forschungen zum SOPHIST-REgelwerk

Was gibt es in der aktuellen Literatur für Hinweise und Meinungen zum SOPHIST-Regelwerk?

In einigen deutschsprachigen wissenschaftlichen Arbeiten und in Büchern zum Themengebiet Requirements Engineering wird auf die Artikel und Bücher zum SOPHIST-REgelwerk verwiesen. Als Beispiele können hier die Dissertationen bzw. Diplomarbeiten von (Kof05), (Ryser03), (Havemeister03) oder (Richter98) gelten.[31]

In den untersuchten Arbeiten wird das Regelwerk bewertungsneutral als Alternative zu eigenen Forschungsarbeiten vorgestellt. Lediglich (Richter98) kommt in seiner Arbeit zu dem Ergebnis, dass das Regelwerk aufgrund des Zeitbedarfs nur für kleinere Projekte geeignet ist.[32] Die Zahl der Verweise speziell auf das Regelwerk kann als vergleichsweise gering angegeben werden. Zwar wird häufig (Rupp07) bzw. die vorhergehenden Auflagen zitiert, außerdem ist die SOPHIST-Group nach eigenen Angaben Marktführer im

Geringe Verbreitung

[31] Vgl. (Kof05), (Ryser03), (Richter98), (Havemeister03)
[32] Vgl. (Richter98), S. 60

Bereich des Requirements Engineerings, das Regelwerk selbst tritt in fremden Publikationen jedoch eher wenig in Erscheinung.

6.5 Ausschlusskriterien für die Anwendung

Gibt es Kriterien, die den Einsatz des SOPHIST-Regelwerks in einem Projekt ausschließen oder verbieten?

Die bis zu diesem Punkt beantworteten Fragen beschäftigen sich damit, wie hoch der Aufwand beim Einsatz des Regelwerks ist und ob das Regelwerk akzeptiert würde. Hier ist klar zu erkennen, dass ein ausreichendes Budget für den Einsatz des Regelwerks nötig ist. Ferner wurde bereits angesprochen, dass das Regelwerk eher für kleinere bis mittlere Projekte geeignet ist.

Neben den Wirtschaftlichkeits-Aspekten existieren jedoch eine Reihe weiterer Punkte, die den Einsatz des Regelwerks erschweren oder unmöglich machen und die im Nachfolgenden aufgezeigt werden sollen:

Ausschreibung

Das Interesse an klaren Anforderungen besteht bereits vor der Auftragsvergabe. Soll auf Basis einer Ausschreibung (die im Normalfall bereits grobe Anforderungen enthält) ein Angebot abgegeben werden, ist es notwendig, dass diese Anforderungen eindeutig interpretiert werden. Es besteht zwar hier meist die Möglichkeit Rückfragen zu stellen, jedoch nicht in dem für das Regelwerk benötigten Umfang. Das Regelwerk kann also in dieser Phase bestenfalls benutzt werden, um Risiken aus unklaren Anforderungen besser zu erkennen.

Rückfragemöglichkeit

Wurde die Durchführung eines Entwicklungsprojekts beschlossen und vergeben, ist es für die Anwendung des Regelwerks weiter unumgänglich, dass der Systemanalytiker Rückfragen beim Anforderungssteller stellen kann. Hierzu ist ein zeitlich verfügbarer und fachlich ausreichend kompetenter Anforderungssteller nötig.

Bereitschaft

Dieser muss ferner bereit sein, genaue Anforderungen abzugeben. Oftmals existiert hier eine Hemmschwelle, da präzise Anforderungen den Verhandlungsspielraum bei nachträglichen Änderungen oder beim Erkennen von Fehlanforderungen erheblich einengen.

Als weiteres Ausschlusskriterium mögen rasch wechselnde Anforderungen gelten. Hier ist der Aufwand für präzise formulierte Anforderungen zu hoch, so dass andere Strategien wie Prototypenentwicklung oder ein agiles Vorgehen gewählt werden müssen.

Agiles Vorgehen

Schließlich muss auch die korrekte Weiterverarbeitung der Anforderungen sichergestellt werden. Der Aufwand in der Erhebung kann nur gerechtfertigt sein, wenn im späteren Verlauf die Entwickler und Tester dem Wortlaut der Anforderungen folgen. Sind diese es jedoch gewohnt, Anforderungen frei zu interpretieren (etwa weil diese gewöhnlich in schlechter Qualität abgeliefert werden) muss zunächst ein Umdenken stattfinden. Ansonsten besteht die Gefahr, dass der Aufwand für die Erstellung der Anforderungen ungenutzt verpufft und letztlich ein System entwickelt wird, das so gar nicht spezifiziert wurde.

Korrekte Weiterverarbeitung

Zusammenfassend kann also gesagt werden, dass vor dem Einsatz in der Praxis die organisatorischen Notwendigkeiten geprüft bzw. geschaffen werden müssen. Dies kann nur auf gemeinschaftlicher Basis zwischen den Projektbeteiligten geschehen, damit der Einsatz des Regelwerks akzeptiert und verstanden wird.

6.6 Fehlertoleranz des Regelwerks

Wie tolerant ist das SOPHIST-REgelwerk im Bezug auf Fehler bei seinem Einsatz? Was passiert, wenn Regeln vergessen/übersehen werden?

Es ist davon auszugehen, dass durch das Vergessen oder Übersehen von Regeln auf jeden Fall ein Qualitätsverlust bei der betroffenen Anforderung entsteht. Die Qualität der Anforderung ist aber immer noch besser, als wenn das Regelwerk gar nicht angewendet würde.

Qualität trotzdem besser wie ohne Regelwerk

Des Weiteren ist zu bemerken, dass die Regeln unterschiedlich wichtig sind und somit auch ein Vergessen unterschiedlich schwere Auswirkungen auf die Qualität der Anforderung hat. So ist die Auswirkung wenn zum Beispiel Regel 1 „Formulieren Sie Anforderungen im Aktiv" vergessen wird „schlimmer" als wenn Regel 10 „Verwenden Sie nur definierte quantitative Angaben" übersehen wird. Im ersten Fall wird möglicherweise die Information

Regeln haben unterschiedliche Auswirkungen

wer eine Tätigkeit ausführt vergessen und es kommt zu späteren Unklarheiten. Im zweiten Fall ist die Auswirkung geringer, da es hier hauptsächlich um die eindeutige Formulierung der Anforderung geht, als dass fehlende Informationen aufgedeckt werden sollten. Als Beispiel soll hier eine Anforderung aus (Rupp07) verwendet werden:

Richtige Anforderung: *„Das System soll jedem Gast jederzeit die Funktionalität zur Verfügung stellen, eine Sicherung aller aufgezeichneten Daten, die im System über ihn gespeichert sind, auf USB-Stick zu initiieren.“*[33]

Fehlerhafte Anforderung: *Das System soll dem Gast die Funktionalität zur Verfügung stellen, eine Sicherung der aufgezeichneten Daten, die im System über ihn gespeichert sind, auf USB-Stick zu initiieren.*

Es zeigt sich, dass die fehlerhafte Anforderung durchaus verständlich ist und es ist nicht erkennbar, wie die Nichtanwendung der Regel 10 zu späteren Problemen führen könnte.

Zur weiteren Verdeutlichung der Problematik kann der Vergleich zwischen Regel 7 „Finden Sie implizite Annahmen“ und Regel 19 „Vermeiden Sie Redundanzen“ dargestellt werden. Wird Regel 7 an einer entscheidenden Stelle vergessen, so kann es sein, dass ganze Sachverhalte komplett vergessen werden. In der Simulation hätte so zum Beispiel das Verhältnis des Reisebuchungssystems zum CRS-Server nicht aufgedeckt werden können – für das System hätte sich ein komplett falsches Bild ergeben. Wenn hingegen Regel 19 vergessen wird, so ist eine Anforderung lediglich zu „blumig“ in ihrer Beschreibung.

Anfänglich häufiges Vergessen

Es kann außerdem behauptet werden, dass zu Beginn der Anwendung des Regelwerks häufiger Regeln übersehen werden, da dem Anwender die Erfahrung fehlt und er das Regelwerk nicht intuitiv anwenden kann.

Test des Vergessens

Um die Fehlertoleranz des Regelwerks zu bewerten wird hier nun an einer Beispielanforderung aus der Simulation getestet, wie sich die Ergebnisanforderung von der eigentlichen Ergebnisanforderung unterscheidet, wenn Regeln vergessen werden.

---

[33] (Rupp07), S. 159.

Ausgangsanforderung ist die Anforderung 10:

> *Es gibt eine Detailanzeige, die dem Kunden dann genauere Informationen zu dem Hotel geben kann.*

Anwendbare Regeln: In der Simulation wurden die Regeln 14 „Machen Sie konkrete Angaben über die Menge an Objekten‟ und die Regel 21 „Überprüfen Sie Nebensätze‟ angewendet.

Hier soll nun die Regel 21 „vergessen‟ werden.

> *Neue fehlerhafte Anforderung: Das System stellt jedem Kunden genau eine Detailanzeige zu jedem Hotel bereit, die dem Kunden genauere Informationen zu dem Hotel gibt.*

Der Unterschied zur Ergebnisanforderung der ersten Iteration ist an dieser Stelle noch nicht bemerkenswert. Durch das Nichtaufdecken des Nebensatzes als Kommentar wird die Anforderung lediglich etwas länger, aber es fehlen noch keine entscheidenden Informationen.

Auf diese erste, neue Anforderung lassen sich dann die Regeln 4 „Ermitteln Sie unvollständige Vergleiche und Steigerungen‟, 5 „Klären Sie Mögliches und Unmögliches‟, 7 „Finden Sie implizite Annahmen‟, 8 „Überprüfen Sie das Verhältnis der Objekte‟ und 12 „Hinterfragen Sie Substantive ohne Bezugsindex‟ anwenden.

Hier ist schon erkennbar, dass durch das Vergessen von Regel 21 im ersten Schritt hier nun eine Regel mehr – nämlich Regel 12 – zum Einsatz kommt. Normalerweise hätte diese Regel hier gar nicht mehr greifen können, da der Sachverhalt „genauere Informationen‟ bereits über die Regel 21 als Kommentar gestrichen worden wäre.

Im zweiten Schritt soll nun die Regel 5 „vergessen‟ werden.

Endgültige fehlerhafte Anforderung:

> *Das System stellt jedem Kunden genau eine Detailanzeige zu jedem Hotel bereit, die dem Kunden Detailinformationen zu jedem Hotel gibt.*

Endgültige richtige Anforderung:

> *Das System stellt jedem Sachbearbeiter genau eine Detailanzeige mit allen Detailinformationen zu jedem Hotel aus der Ergebnisliste für jeden Kunden zur Verfügung.*

Es wird deutlich, dass die fehlerhafte Anforderung nur geringe Unterschiede zur richtigen Anforderung aufweist. Was allerdings aus dieser Anforderung

nicht ersichtlich wird ist, dass nicht geklärt wurde, dass die Detailinformationen vom CRS-Server stammen, dass die Detailinformationen in einer neuen eigenen Anzeigemaske dargestellt werden sollen und dass die Informationen als Fließtext angezeigt werden sollen. Diese Sachverhalte wurden bei Anwendung der Regel durch die Aufstellung einer neuen zusätzlichen Anforderung abgedeckt.
Bei der Betrachtung der Umwandlung in der Simulation wird aber auch deutlich, dass selbst wenn die Regel 5 vergessen worden wäre, der Sachverhalt bezüglich des CRS-Servers aufgrund der Regeln 7 und 8 trotzdem noch aufgedeckt worden wäre.

Fehler werden trotzdem aufgedeckt – Qualität besser

Verschiedene Regeln decken somit grobe Fehler und Versäumnisse über unterschiedliche Herangehensweisen und Fragestellungen doch noch auf. Wie schon zuvor erwähnt, ist somit die Qualität der Anforderungen schlechter als wenn alle Regeln beachtet würden, aber auf jeden Fall deutlich besser, als bei einem Verzicht auf die Anwendung des SOPHIST-REgelwerk.

6.7 Andere Methoden in der Systemanalyse

6.7.1 Vergleich zu anderen Methoden

In der Systemanalyse gibt es verschiedenste Methoden zur Erhebung von Anforderungen. Hier soll nun kurz ein Überblick über die Methodengruppen gegeben werden. Anschließend soll näher auf die Befragungstechniken eingegangen werden, zu denen auch das SOPHIST-REgelwerk zählt.

Es gibt zurzeit vier verschiedene Ermittlungstechnik und zusätzlich unterstützende Techniken. Zu den vier Ermittlungstechniken gehören die Kreativitätstechniken, die Beobachtungstechniken, die Befragungstechniken, sowie die vergangenheitsorientierten Techniken.

Ermittlungstechniken

Unter den Kreativitätstechniken versteht man Techniken wie zum Beispiel das Brainstorming. Hierbei wird versucht, so kreativ wie möglich zu sein um auch neue Ideen zu entwickeln. Bei den Beobachtungstechniken liegt der Fokus auf der Beobachtung des Anforderungsstellers bei seiner Arbeit. Es wird versucht aus seinen Arbeitsabläufen die Anforderungen zu ermitteln. Die zwei Methoden dieser Gruppe sind die Feldbeobachtung und das Ap-

prenticing. Die älteste Gruppe der Erhebungstechniken sind die Befragungstechniken. Der Anforderungssteller wird hier direkt befragt. Zu dieser Gruppe gehört, wie schon erwähnt, das SOPHIST-REgelwerk. Die Systemarchäologie und der Reuse gehören zu den Vergangenheitsorientierten Techniken. Betrachtet wird hier das alte System beziehungsweise die zugehörigen Dokumente wie das Benutzerhandbuch und ähnliches.[34]

Unterstützende Techniken

Die unterstützenden Techniken versuchen die oben genannten Techniken zu unterstützen. Zu diesen Techniken zählen unter anderem das Mind Mapping, die Durchführung von Workshops aber auch die Aufzeichnung der Anforderungserhebung durch Video oder Audio.[35]

Befragungstechniken

Da das SOPHIST-REgelwerk zu den Befragungstechniken gehört, soll nun näher auf diese Gruppe eingegangen werden. Wie schon erwähnt werden hierbei der / die Anforderungssteller direkt befragt. Die einzelnen Methoden sollen Hilfestellung beim Stellen der Fragen geben um so das explizite Wissen der Anforderungssteller geschickt abfragen zu können.
Die folgende Tabelle gibt hierfür einen Überblick:

Methode	Grundlage	Anwendung	Vorteile	Nachteile
SOPHIST-REgelwerk	Neuro-Linguistische-Programmierung	Verbesserung der Qualität von bereits aufgenommenen Anforderungen; Überprüfung der Vollständigkeit	Qualitätsverbesserung; Vermeidung von Lücken und Missverständnissen	viel Erfahrung nötig
Fragebogen	Fragebogen mit Multiple-Choice-Fragen	gut geeignet bei der Produktentwicklung;	Befragung vieler Stakeholder; geringer Zeit- / Kostenaufwand	schlecht zur Abfrage von impliziten Wissen; Rückfragen sehr aufwändig
Interview	Stellen von Fragen und Protokollierung der Antworten	persönliche Interviews oder elektronisch; einsetzbar in der gesamten Systementwicklung	individuelle Anpassung an das Gespräch; gezieltes Nachfragen	sehr zeitaufwändig je mehr Anforderungssteller
Selbstaufschreibung	Verfassen von Tätigkeitsbeschreibungen durch Stakeholder	zur Ermittlung bestehender Arbeitsabläufe	wenig Zeitaufwand für den Analytiker	Dokumentation nur der bewussten Anforderungen; stark abhängig von der Motivation der Stakeholder
On-Site-Customer	ein Stakeholder	Ermittlung der Anforderungen auf	schnelle Ermittlung der Anfor-	sehr hoher Aufwand für den

[34] Vgl. (Rupp07), S. 115ff.
[35] Vgl. (Rupp07), S. 128ff.

	ständig beim Entwickler-Team verfügbar	schnelle, effiziente und unbürokratische Weise	derungen, geeignet für eXtreme Programming	Kunden; der Stakeholder muss sich ständig mit den anderen Stakeholdern abstimmen

Tabelle 6: Überblick über Befragungstechniken
(Quelle: (Rupp07), S. 122 ff.)

6.7.2 Kompatibilität zu anderen Methoden

Kann das SOPHIST-REgelwerk im Zusammenhang mit anderen Methoden eingesetzt werden?

Einsatz bei allen dokumentierten Anforderungen

Da das SOPHIST-REgelwerk auf dokumentierte Anforderungen angewendet wird um hier die Qualität der Anforderungen zu verbessern und um möglicherweise aufgetretene Missverständnisse aufzudecken, kann es mit jeder anderen Methode, aus der textuelle Anforderungen hervorgehen eingesetzt werden.

Es kann somit zusammen mit anderen Befragungstechniken wie dem Interview oder der Selbstaufschreibung verwendet werden. Durchaus ist aber auch eine Überprüfung der Anforderungen denkbar, die zum Beispiel durch ein Brainstorming oder eine Beobachtungstechnik erhoben wurde.

Voraussetzung für Kombination: schriftliche Anforderungen

Voraussetzung für die Kombination des SOPHIST-REgelwerks mit einer anderen Methode ist allerdings, dass aus der jeweiligen Methode schriftliche Anforderungen resultieren. Stichwortartige Skizzierungen der Ideen – wie beispielsweise beim Mind Mapping – reichen nicht aus um die Regeln anwenden zu können, da hier die Voraussetzungen (Satzbau, Beschreibungen...) für die Regeln fehlen.

6.7.3 Überführbarkeit in andere Methoden

Wie gestaltet sich die Überführung der Anforderungen nach SOPHIST zum Beispiel in ein UML-Modell?

Methodenneutralität

Zu diesem Punkt lässt sich zunächst feststellen, dass das SOPHIST-REgelwerk grundsätzlich unabhängig von einer Modellierungssprache oder einem Anwendungssystemtyp ist. Zwar wird in (Rupp07) der Einsatz der UML für Weiterverarbeitung und Dokumentation der Anforderungen vorgeschlagen, eine Ausrichtung oder Unterstützung auf die Sprachbestandteile

der UML ist jedoch nicht gegeben. Der Einsatz anderer domänenunabhängiger oder domänenspezifischer Modellierungssprachen ist somit möglich.

Keine Unterstützung einer Modellierungssprache

Mit der Unabhängigkeit von einer Modellierungssprache geht auch die fehlende Unterstützung einer solchen einher. Im Gegensatz zu vielen Forschungsansätzen aus dem Bereich Requirements Engineering bleibt es im SOPHIST-REgelwerk die alleinige, intellektuelle Tätigkeit des Systemanalytikers für die textuellen Anforderungen entsprechende Modelle zu erzeugen. Eine Unterstützung, um bspw. potentielle Klassen oder Attribute zu erkennen, gibt es nicht.

6.8 Möglichkeiten der Unterstützung

6.8.1 Unterstützungstools

Gibt es Tools, die den Einsatz des SOPHIST-Regelwerks unterstützen?

Folgende Möglichkeiten lassen sich erkennen, um den Einsatz des Regelwerks zu unterstützen:

- Unterstützung bei der Dokumentation von Rückfragen und der Umformulierung von Anforderungen (z. B. mit Textbausteinen).
- Interaktive Führung durch das Regelwerk mittels Vorgabe von Fragen bzw. Markieren von „verdächtigen" Formulierungen in Anforderungen.
- Halb- bis Vollautomatische Prüfung der Regeln des Regelwerks und Korrektur der Fehler über ein Werkzeug.

CARE und CARA

Für die erste Variante der Unterstützung kann CARE, das Requirements Management Tool der SOPHIST-Group, aufgeführt werden. Es unterstützt das Vorbereiten von Interviews über die Möglichkeit Fragen zu Anforderungen zu dokumentieren.

Für die Varianten zwei und drei ist derzeit kein kommerziell verfügbares Werkzeug erhältlich. Die SOPHIST-Group hat zwar mit CARA an einem Werkzeug gearbeitet, dass die teilautomatische Prüfung englischer Anforderungen ermöglicht und damit der zweiten oder dritten Gruppe zuzuordnen wäre. Nach Angaben der SOPHIST-Group erfuhr diese Anwendung jedoch keine weitere Entwicklung über den Prototypen-Status hinaus. CARA ist zu-

dem sehr auf das Erkennen von Signalwörtern fixiert, deren Brauchbarkeit bereits an früherer Stelle angezweifelt wurde.

Welche Tools eignen sich für die Verwaltung von Anforderungen, die über das SOPHIST-Regelwerk bearbeitet wurden?
Nur rein für die Dokumentation und Verwaltung von Anforderungen, ist der Einsatz eines Werkzeuges zu empfehlen.

Standard Office Produkte

Selbstverständlich können die Anforderungen, die mit dem SOPHIST-REgelwerk bearbeitet wurden, über die Standard-Produkte von Microsoft wie Word oder Excel verwaltet werden. Diese Programme bieten allerdings nicht viel Unterstützung bei der Verwaltung der Anforderungen für den Anwender an.

RM-Tools wie CARE

Mit den heute gängigen Requirements Management Tools werden sich die Anforderungen nach dem SOPHIST-REgelwerk gut verwalten lassen. Empfehlenswert wäre der Einsatz des Tools CARE (Computer Aided Requirements Engineering), da es sich bei diesem Tool um eine Entwicklung der SOPHIST-Group selbst handelt. Es unterstützt besonders die frühen Projektphase – also das Sammeln, Optimieren und Verwalten von Anforderungen – und es ist mit diesem Tool möglich Anforderungen direkt zum Beispiel aus einem Interview-Protokoll zu übernehmen.
Für die Verwaltung der Anforderungen nach Requirements Management Aspekten können aber auch Tools wie DOORS, Caliber RM oder RequisitePro eingesetzt werden. Hier sollte eine Abwägung der einzelnen Systemeigenschaften erfolgen um das beste Produkt für den jeweiligen Einsatzbereich zu finden.

6.8.2 Andere Unterstützungsarten

Gibt es Infomaterial von der SOPHIST Group, die das SOPHIST Regelwerk etwas näher erläutern, als die Beschreibung im Buch von Chris Rupp?
Es gibt vergleichsweise wenige Veröffentlichungen zum SOPHIST-REgelwerk von Chris Rupp oder der SOPHIST-Group. Als „Hauptwerk" kann weiterhin das Buch „Requirements Engineering und Management" in der jeweils aktuellen Auflage bezeichnet werden, das den Stand des Regelwerks wiedergibt.

Auf www.sophist.de ist zudem eine Ausarbeitung älteren Datums zum Thema zu finden, sowie einige eher allgemeine Artikel aus Fachzeitschriften. Ebenso sind einige zusammenfassende Präsentationen zum Regelwerk im Internet zu finden.
Die Informationsquellen zum Regelwerk sind damit eher gering und gleichen sich sehr stark. Dies ist insofern problematisch, da in der ausführlichsten Darstellung des Regelwerks in (Rupp07) weiterhin Fragen ungeklärt bleiben (siehe Kap. 4.1.2). Da diese auch in den überblicksartigen Darstellungen nicht geklärt werden, muss an einigen Stellen selbst recherchiert bzw. interpretiert werden.

6.9 Zusammenfassung

Die Bewertung der Praxisrelevanz zeigte, dass erhebliche Zweifel vorhanden sind, ob das Regelwerk in Projekten eingesetzt werden kann.
Vor allem der zeitliche Aufwand wurde als großes Hindernis identifiziert, aber auch die Akzeptanz auf Seiten von Systemanalytikern und Anforderungsstellern wurde nicht durchgehend positiv bewertet. Zudem wurden eine Reihe weiterer organisatorischer Ausschlusskriterien bestimmt.
Das geringe Schulungsangebot zum Regelwerk und die geringe Verbreitung in der Literatur, lassen zudem auf eine eher geringe Bedeutung in der Praxis schließen.
Das Projektteam kam zu dem Schluss, dass das SOPHIST-REgelwerk nur unter bestimmten Rahmenbedingungen (hohes Schadenspotential, ausreichende Zeit, ausreichendes Budget, hohe Ausgangsqualität, begrenzter Umfang) einer Nutzenanalyse standhalten wird.
Als Alternative wurde angeboten, den Detaillierungsgrad erheblich zu mindern oder sich auf die Anwendung einiger Regeln zu beschränken. Bereits aus einer Teilanwendung dürfte der Entwicklungsprozess profitieren. Der geringe Einarbeitungsaufwand und die hohe Fehlertoleranz des Regelwerks wirken hier unterstützend.

7 Bewertung der Relevanz für die Lehre

Der letzte Teil der Bewertung klärt die Frage, ob das SOPHIST-REgelwerk in die Lehre an einer Berufsakademie aufgenommen werden sollte. Es werden nicht nur die Gründe behandelt die für bzw. gegen eine Aufnahme sprechen, sondern auch gezeigt, wie eine Veranstaltung zum Thema sinnvollerweise aussehen könnte.

7.1 Erlernbarkeit des Regelwerks in der Lehre

Kann das SOPHIST-REgelwerk so aufbereitet werden, dass die Studenten es in einer Vorlesung erlernen können?

Rein theoretische Vorstellung nicht sinnvoll.

Betrachtet man die Erfahrungen aus der Simulation, so wird deutlich, dass eine rein theoretische Behandlung des Regelwerks nicht zielführend sein kann. Würden die 25 Regeln ohne Übung vorgestellt, so ist davon auszugehen, dass kein vertieftes Verständnis zu erlangen ist. Auch die Unterlegung der Regeln mit treffenden Beispielen erreicht nur einen geringen zusätzlichen Lerneffekt.

Alternative Lehrmethoden bringen Erfolg

Es wäre empfehlenswerter, das Regelwerk mit Hilfe von interaktiven Lehrmethoden vorzustellen. So sollte den Studenten die Möglichkeit gegeben werden, das SOPHIST-REgelwerk selbst zu testen und auf Beispielanforderungen anzuwenden. Sie können so durch die praktische Anwendung erkennen, wie das Regelwerk eingesetzt wird, welchen Aufwand der Einsatz des Regelwerks darstellt und wo Unklarheiten entstehen. Damit lässt sich ein Verständnis dafür entwickeln, was eine qualitativ hochwertig formulierte Anforderung ausmacht.

Ist es sinnvoll das Regelwerk auswendig zu lernen?

Im Kapitel 4.1.1 wurde schon auf die intuitive Anwendung des SOPHIST-REgelwerks eingegangen und es wurde erläutert, dass das Regelwerk nicht komplett intuitiv angewendet werden kann. Ferner wurde als nötiges Hilfsmittel eine Auflistung der Regeln bestimmt, die die Vollständigkeit der Anwendung garantiert.

Im Zusammenhang mit der Lehre stellt sich erneut die Frage, ob das Regelwerk von den Studenten auswendig gelernt werden sollte.

Aufgrund der Erfahrungen aus der Simulation ist von dem Auswendiglernen der 25 Regeln abzuraten, da die Regeln im späteren Praxiseinsatz auch nicht unbedingt auswendig beherrscht werden müssen. Es ist wichtiger, zu vermitteln, was die einzelnen Regeln beinhalten und welche Effekte sie aufdecken.

Auswendiglernen nicht sinnvoll

7.2 Überlegungen zur Didaktik

7.2.1 Notwendiges Vorwissen

Welches Vorwissen oder welcher Wissensrahmen ist nötig um das SOPHIST-REgelwerk zu verstehen?

Um das Regelwerk anwenden zu können, sind grammatikalische Kenntnisse, sowie in geringem Umfang Wissen über Neuro-Linguistische-Fachbegriffe nötig. Beides kann im Rahmen der Lehrveranstaltung vermittelt werden. Gerade das grammatikalische Grundwissen sollte bei Studenten aus der meist noch nicht lange zurückliegenden Schulzeit noch in ausreichendem Maße präsent sein.

Soll auch die Dokumentation von Anforderungen behandelt werden, sollten zumindest UML Use-Case-Diagramme bekannt sein. Als einfachster UML-Diagrammtyp dürfte auch dieser in kurzer Zeit zu vermitteln sein. Vertieftes Wissen über Modellierungssprachen ist wegen der Methodenneutralität des Regelwerks nicht nötig.

Weitere Kenntnisse

Als positiv können sich bereits gemachte Erfahrungen aus einer Anforderungserhebung in der Praxis für Studenten erweisen. Dadurch wird der Zugang zum Thema erleichtert und eine Motivation für mögliche Verbesserungen des Vorgehens geschaffen. Auch kann die Praxistauglichkeit damit besser eingeschätzt werden. Eine zwingende Voraussetzung stellen Erfahrungen aus diesem Bereich jedoch nicht dar.

Erfahrung

Da damit kein Wissen aus höheren Semestern nötig ist und die Konfrontation mit der Aufgabenstellung einer Anforderungserhebung in den betrieblichen Praxisphasen jedoch meist frühzeitig stattfindet, spricht nichts dagegen, das Regelwerk bereits im ersten oder zweiten Semester zu lehren.

Frühzeitige Anwendung

7.2.2 Prüfbarkeit des Lernerfolgs

Kann der Lernerfolg auch abgefragt werden? Wie sollten mögliche Kontrollfragen aussehen?

Seminararbeiten

Zum einen könnten die Studenten im Rahmen von Seminararbeiten verschiedene Anforderungen bearbeiten und die Anwendung des SOPHIST-REgelwerks näher beschreiben. Die Anforderungen für die Seminararbeiten könnten für jeden Studenten einzeln vorgegeben werden, so dass jeder Student dann eigenständig darstellen muss, wie er die einzelnen Regeln interpretiert und anwendet. Vorteil ist, dass hierbei eine intensive Beschäftigung mit dem Thema ohne großen Zeitdruck möglich ist.

Eine weitere denkbare Möglichkeit wäre auch, dass die Studenten in kleineren Gruppen eine Anforderungserhebung durchführen und dann das SOPHIST-REgelwerk auf die erhobenen Anforderungen anwenden. In der Seminararbeit wären dann das Vorgehen bei der Erhebung sowie die Anwendung des Regelwerks mit entsprechenden Begründungen zu dokumentieren. Dies entspräche in etwa der durchgeführten Simulation. Da der Aufwand für dieses Vorgehen sehr hoch ist, kann dies nicht als reguläre Prüfungsmöglichkeit angesehen werden.

Klausuraufgabe

Soll die Leistungskontrolle durch eine Klausur erfolgen, so ist – wie schon erläutert – das Auswendiglernen der einzelnen Regeln nicht sinnvoll. In der Klausur sollten die Studenten stattdessen zeigen können, dass sie das SOPHIST-REgelwerk und dessen Anwendung verstanden haben. Es könnten hierzu Beispielanforderungen gestellt werden, die dann – unterstützt durch eine Übersichtsliste mit allen Regeln – nach dem SOPHIST-REgelwerk zu bearbeiten sind. Natürlich können entstehende Rückfragen nicht geklärt werden. Hier wäre es dann ausreichend, wenn die Studenten die verursachende Regel sowie die entstehenden Rückfragen notieren.

7.3 Vorteile für den Studenten

Für einen Studenten hat die Teilnahme an einer Veranstaltung zum SOPHIST-REgelwerk entscheidende Vorteile, wenn er im späteren Berufsleben als Systemanalytiker oder Entwickler tätig sein wird und Anforderungen analysieren muss. In diesen Fällen profitiert er zwangsläufig von dem Wissen über mögliche Fehlerquellen, selbst wenn er das Regelwerk im Betrieb nicht explizit auf Anforderungen anwendet.

Vorteile sind aber auch für andere Tätigkeitsfelder zu erkennen. So kann ein Projektleiter mit Kenntnis des Regelwerks Risiken beim Lesen von Anforderungen besser abschätzen. Darüber hinaus ist es oft der Fall, dass Wirtschaftsinformatiker als Anforderungssteller auftreten, wenn bspw. Projekte vergeben werden (Koordinatorenfunktion). Auch hier ist das Wissen aus dem Regelwerk nützlich.

7.4 Relevanz für die Praxis

Wie wirkt sich die Praxisrelevanz auf die Relevanz in der Lehre aus?

In der Bewertung der Praxisrelevanz wurde erkannt, dass das Regelwerk nur unter sehr bestimmten Randbedingungen (Budget, Projektumfang etc.) wirtschaftlich anwendbar ist und es Ausschlusskriterien (Mitarbeit des Fachbereichs etc.) gibt, die häufig in Betrieben vorkommen. Aber auch wenn diese nicht gegeben sind, ist es sicher nicht immer einfach die gängige Praxis zu ändern.

Geringer Einfluss

Diese Schranken sollten in der Bewertung der Lehrrelevanz nicht überbewertet werden. Allein das Wissen über die Grundprinzipien des Regelwerks kann im Einzelfall auch ohne explizite Anwendung des Regelwerks positive Auswirkungen haben. Ferner sollte sich jede Hochschule als Innovationstreiber verstehen und ihren Studenten neue Methoden vermitteln. Ist eine ausreichende Menge an Fachkräften vorhanden, die die Methode kennen und unterstützen, lassen sich auch die nötigen organisatorischen Änderungen leichter durchsetzen, damit diese angewendet werden. Im Fall des SOPHIST-REgelwerks würde dies eine schon seit langem propagierte Verbesserung der Systemanalyse zur Folge haben.

7.5 Mögliche Widerstände

Ist das Regelwerk wissenschaftlich ausreichend begründet?

Zweifelhafte wissen-schaftliche Hinter-legung

Bei der Lektüre von (Rupp07) mag der Eindruck entstehen, dass die Verbindung von Requirements Engineering und Neuro-Linguistischen-Programmieren nur konstruiert ist. Gerade die Unbrauchbarkeit von Signalwörtern, aber auch der Umstand, dass bestimmte Regeln in jeder Anforderung anwendbar sind, lassen vermuten, dass zwischen einem Therapiegespräch (NeuroLinguistisches-Programmieren) und Anforderungen (SOPHIST-REgelwerk) zu große Unterschiede bestehen als dass beide mit den gleichen Theorien behandelt werden können. So ist bspw. davon auszugehen, dass in einem Therapiegespräch nicht in jedem Satz eine Möglichkeit ausgedrückt wird und daher eine Regel, die dies explizit untersucht, Sinn ergibt. In Anforderungen, in denen jeder Satz eine Möglichkeit ausdrückt, erscheint eine explizite Prüfung jedoch überflüssig.

Die willkürliche Gliederung in Tilgung, Generalisierung und Transformation unterstützt diesen Eindruck. Hier wären gerade bei Anforderungen, die bestimmten Mustern folgen, Oberpunkte wie „Objekte", „Prozesse", „Qualitätskriterien" o. ä. einleuchtender.

Sieht man über diese Punkte hinweg, erweisen sich die Regeln jedoch als gut anwendbar und liefern gute Ergebnisse. Auch wenn ihre wissenschaftliche Begründung nicht unumstritten ist, können sie somit zumindest als Best-Practise-Ansatz gelten.

7.6 Aufwand für die Aufbereitung

Wie viel Zeit braucht ein Dozent um das SOPHIST-REgelwerk für seine Vorlesung aufzubereiten?

Vorbereit-ung für eine Vorlesung

Soll das SOPHIST-REgelwerk ausschließlich in der Vorlesung durch eine Präsentation vorgestellt werden, so kann der Dozent von einem Vorbereitungsaufwand von ca. 5 – 6 Stunden ausgehen. Hier ist mit eingerechnet, dass der Dozent etwa zwei Stunden für die Lektüre und die Einarbeitung in das Regelwerk und das Umfeld benötigt. Die restliche Zeit wird zur Vorbereitung der Folien angesetzt. Dieser Aufwand ist etwas höher, da sich der Dozent eventuell noch eigene Beispielanforderungen zu den einzelnen Anforderungen überlegen muss, die er in der Vorlesung mit den Studenten bearbeiten kann.

Je nach Hintergrund und Vorwissen des Dozenten ist unter Umständen noch Aufwand für die Klärung der Fachbegriffe einzurechnen. Hier kann von zwei weiteren Stunden ausgegangen werden.

Fachbegriffe klären

Sollte der Dozent Übungen für die Erarbeitung des SOPHIST-REgelwerks für die Studenten vorsehen, so entsteht hier – natürlich abhängig von der Art der Übung – ein weiterer zeitlicher Aufwand (dieser kann hier aufgrund der Vielfalt von möglichen Übungen nicht näher spezifiziert werden).

Vorbereitung von Übungen

Als Übungen könnte der Dozent beispielsweise „schlechte" Anforderungen an die Studenten austeilen und nach den Regeln des SOPHIST-REgelwerks bearbeiten lassen. Verfügt der Dozent aus seiner praktischen Arbeit oder aus anderen Quellen bereits über einen Katalog von derartigen Anforderungen, so fällt für ihn der zeitliche Aufwand für das Erstellen von „schlechten" Anforderungen entsprechend geringer aus.

7.7 Möglichkeiten der Darstellung

Kann das SOPHIST-REgelwerk so aufbereitet werden, dass es für die Studenten in der Vorlesung verständlich ist? Wie sollte das SOPHIST-REgelwerk dargestellt werden, um es optimal zu vermitteln?

Um das SOPHIST-REgelwerk im Rahmen einer Vorlesung für Studenten angemessen zu vermitteln, muss der Inhalt des Regelwerks zunächst in seiner Gesamtheit in der Theorie dargestellt werden. Dazu kann die Gliederung aus (Rupp07) verwendet werden. Die bemängelten Punkte in (Rupp07) mit unklaren bzw. nicht zuordenbaren Beispielen sollten jedoch nicht übernommen werden, sondern vielmehr eine klare Abgrenzung der Beispiele erfolgen. Wie auch das Projektteam muss der Dozent dazu eine eindeutige Interpretation des Regelwerks für sich vornehmen.

Allgemeines Lehren des Regelwerks

Nach einer theoretischen Einführung erscheint es dringend notwendig, den Studenten in Form einer Übung oder eines Workshops die Möglichkeit zu geben das Regelwerk selbst anwenden. So wird gewährleistet, dass diese das Regelwerk verinnerlichen und dass offene Fragen und Verständnisprobleme zu Tage treten. Ohne eine selbstständige Anwendung, die zumindest im kleinen Rahmen stattfindet, wird es den Studierenden kaum möglich sein, das Zusammenwirken der einzelnen Regeln zu begreifen.

Praktische Umsetzung in der Lehre durch Studierende

Vorgabemöglichkeiten für freie Arbeiten durch Dozenten

Für die Umsetzung des Workshops bzw. der Übung bietet sich die Möglichkeit an, fehlerhafte Anforderungen schriftlich vorzugeben, die die Studenten für sich bearbeiten, oder im Rahmen einer Gruppenarbeit eine Anforderungserhebung ähnlich der Simulation im durchgeführten Projekt- nur in viel kleinerem Rahmen - vorzunehmen. Für die zweite Vorgehensweise spricht, dass diese dem realen Requirements Engineering Prozess als stark kommunikativen Prozess deutlich näher kommt. Dagegen spricht der zeitliche Aufwand, der mehrere Stunden betragen dürfte, wenn er Erhebung und Analyse umfassen soll. In beiden Fällen sollten die Ergebnisse und die entstandenen Fragen im Anschluss besprochen bzw. präsentiert werden.

7.8 Einsatzpunkt in der Lehre

7.8.1 Thematische Zugehörigkeit

In welchem Themenkomplex würde sich die Vorstellung des SOPHIST-REgelwerks anbieten?

Das SOPHIST-REgelwerk lässt sich eindeutig dem Themenkomplex „Systemanalyse“ zuordnen, da die dort behandelten Methoden auf den Ergebnissen einer Anforderungserhebung aufbauen.

Rahmenwerk nötig

Es spricht vieles dafür, das Regelwerk nicht alleinstehend in einer Veranstaltung zu behandeln, sondern es als Teil einer Veranstaltung zum Thema „Requirements Engineering“ zu sehen. Nach den Erfahrungen in der Simulation kann sogar geschlussfolgert werden, dass eine ausschließliche Vermittlung des Regelwerks ohne ergänzende Betrachtung von bspw. Erhebungsverfahren wenig sinnvoll ist.

Eigene Veranstaltung

Für die Etablierung einer eigenen Veranstaltung zum Thema „Requirements Engineering“ spricht auch, dass Veranstaltungen zur Systemanalyse in der Regel bereits sehr umfangreich sind und auch weitere Methoden des Requirements Engineerung wie Erhebungstechniken (Interviews, CRC-Karten etc.), Dokumentationstechniken oder -standards oder die Verwaltung von Anforderungen (Traceability etc.) behandelt werden sollten. Eine eigenständige Veranstaltung „Requirements Engineering“ sollte parallel oder vor einer Veranstaltung „Systemanalyse“ stattfinden.

7.8.2 Zeitaufwand in der Lehre

Wie lange dauert das Vermitteln des Regelwerks?

Die Beantwortung der Frage hängt von der gewählten Lehrmethode ab. So dürfte die theoretische Einführung in 4 Vorlesungseinheiten zu bewerkstelligen sein. Eine Einzelübung auf Basis von schriftlich vorliegenden Anforderungen sollte auf maximal 2 Vorlesungseinheiten begrenzt werden. Eine Gruppenübung mit Erhebung und Analyse bedarf mit Sicherheit der doppelten Zeit. Die Nachbesprechung sollte mit mindestens 2 Einheiten nicht zu kurz ausfallen, da hier aus Fehlern anderer gelernt werden kann und das Regelwerk teilweise zum zweiten Mal angewendet werden kann.

Umfang des Themas in Vorlesungseinheiten

7.8.3 Kombination mit anderen Methoden

Im Vergleich zu welchen anderen Methoden sollte das SOPHIST-REgelwerk vorgestellt werden?

In Kapitel 5.4 wurde dargestellt, wie die Anwendung der Anforderungsschablone aussieht.

Diese Methode scheint für die Sicherung der Qualität von Anforderungen schon bei der Formulierung und der Erhebung unter den dort angegebenen Vorbedingungen sehr geeignet. Das SOPHIST-REgelwerk befasst sich im Gegensatz dazu mit der Verbesserung der Qualität von bereits dokumentierten Anforderungen. Um ein in sich schlüssiges Bild bezüglich der Qualitätsverbesserung bei Anforderungen zu schaffen, sollte sowohl das Vorgehen nach der Anforderungsschablone als auch das SOPHIST-REgelwerk vorgestellt werden.

Vorstellung von Anforderungsschablone

Generell sollten – im Rahmen einer Vorlesung zum Thema „Requirements Engineering“ – verschiedene Methoden zur Erhebung und Ermittlung von Anforderungen (siehe Kapitel 6.7.1) vorgestellt werden. Es sollte das Bewusstsein gefördert werden, dass die Erhebung von Anforderungen bspw. in Form eines Interviews durchaus ihre Schwierigkeiten hat und nicht als Standardübung gesehen werden kann.

Bewusstsein für die Anforderungsermittlung stärken

7.9 Betrachtung anderer Hochschulen

Ist das Regelwerk an anderen Hochschulen präsent?

Über eine Internetrecherche konnten nur wenige Hochschulen ausgemacht werden, in denen das Regelwerk als Teil von Seminaren oder Vorlesungen auftaucht. Wenn es behandelt wird, erfolgt dies dabei teilweise sehr stark zusammengefasst.

Geringe Verbreitung

Als Beispiel kann die TU München dienen, in der das Regelwerk im Rahmen eines Seminars vorgestellt wurde[36]. Ebenso erscheint die Theorie des Regelwerks in der Lehre der ETH Zürich und der FH Bonn.[37] Inwieweit weitere Hochschulen das Regelwerk explizit in Veranstaltungen zu Requirements Engineering behandeln ist schwer bestimmbar, da Vorlesungsunterlagen meist entweder gar nicht oder nur kennwortgeschützt über das Internet zugänglich sind. Es kann jedoch davon ausgegangen werden, dass eine stärkere Präsenz des Regelwerks sich zumindest etwas in den Ergebnissen der Recherche niedergeschlagen hätte. Eine hohe Präsenz des Regelwerks in der Lehre kann derzeit – d. h. im Jahr 2007 - somit ausgeschlossen werden.

7.10 Zusammenfassung

Die Bewertung der Relevanz für die Lehre hat gezeigt, dass das Regelwerk mit angemessenem Aufwand für die Lehre vorbereitet und in einer Veranstaltung vermittelt werden kann.
Dabei wurde betont, dass eine rein theoretische Betrachtung wenig Sinn ergibt und auf jeden Fall mit einer Übung oder einem Workshop kombiniert werden sollte.
Ebenso wurde die Meinung vertreten, dass eine Veranstaltung, die sich rein mit dem Regelwerk beschäftigt für die Lehre an einer Berufsakademie weniger geeignet ist als eine Veranstaltung, die auch weitere Aspekte des Requirements Engineerings wie Erhebungstechniken behandelt.
Dies entspricht auch dem Vorgehen der wenigen anderen Hochschulen, die das Regelwerk in die Lehre aufgenommen haben.

---

[36] Vgl. (TUM07)
[37] Vgl. (IFI07)

Gerade da der Themenbereich „Requirements Engineering“ momentan häufig nur relativ oberflächlich behandelt wird, er aber gleichzeitig eine Kernkompetenz der Wirtschaftsinformatik darstellt, erscheint eine Aufnahme einer speziellen Veranstaltung zu diesem Themenbereich höchst wünschenswert, die dann auch das SOPHIST-Regelwerk beinhalten sollte.

8 Leitfaden für die Anwendung

Als Empfehlung für den Einsatz des Regelwerks wird zunächst das Vorgehensmodell aus Kap. 3.3 vorgeschlagen, in dem davon ausgegangen wird, dass der Anforderungssteller während der Anforderungsanalyse nicht anwesend ist. Es entlastet den Anforderungssteller und ist zudem für Systemanalytiker geeignet, die das Regelwerk erstmalig oder selten anwenden.

Nachfolgend wird basierend auf diesem Vorgehensmodell ein Leitfaden in Form einer Vorlage für die Anforderungsanalyse vorgestellt. Diese Vorlage gibt eine zielgerichtete Reihenfolge für die Regeln vor, präzisiert sie teilweise oder ergänzt sie um Erläuterungen aus (Rupp07).

Die Vorlage versucht sowohl inhaltliche Abhängigkeiten zwischen den Regeln zu berücksichtigen, als auch die konkreten Erfahrungen aus der Simulation aufzugreifen. Inwiefern diese Erfahrungen und damit die Vorlage insgesamt allgemeingültig sind, kann nur durch ihre weitere Erprobung in Erfahrung gebracht werden.

Eine Erweiterung des Regelwerks, bspw. um Regeln, die Abhängigkeiten zwischen Anforderungen behandeln, findet nicht statt. Die Vorlage stellt zudem lediglich einen Entwurf dar, der für die Anwendung in der Praxis möglichst in ein rechnergestütztes Werkzeug integriert werden sollte. Dieses könnte weitere Unterstützung anbieten, indem es bspw. alle Wörter einer bestimmten Wortart markiert oder den Aufbau eines Glossars ermöglicht.

Originalanforderung

Text der Orginalanforderung

Prozesse (Regel 1, 2, 17, 18)

Ziel: Die Anforderung beantwortet die Frage: Wer macht was?

- Untersuchen Sie ob sich die Anforderung im Passiv befindet und überführen Sie diese ggf. ins Aktiv!
- Formulieren Sie die Anforderung dabei so, dass klar wird, welche Aufgabe das System hat („Das System soll …“).
- Untersuchen Sie, ob Prozesse durch Substantive oder Kombinationen aus Substantiven und Verben ausgedrückt werden. Formulieren Sie diese mit Vollverben.
- Ausnahme: Sie können längere Vorgänge zur Komplexitätsreduktion als Substantive ausdrücken, müssen sie dann aber als neue Anforderung(en) oder Glossareintrag genauer spezifizieren. Fügen Sie in dazu einen Verweis ein.

Satzstruktur (Regel 19, 20, 21, 22)

Ziel: Die Anforderung ist knapp und klar.

- Untersuchen Sie, ob mehrere Anforderungen im Satz ausgedrückt werden! Formulieren Sie ggf. neue Anforderungen!
- Untersuchen Sie, ob der Satz eine Begründung enthält! Stellen Sie diese als Kommentar bei!
- Gibt es überflüssige oder floskelhafte Satzbestandteile? Streichen bzw. kürzen sie diese!
- Gibt es eine zeitliche oder logische Abhängigkeit im Satz? Drücken Sie diese über einen „Wenn … dann“ Satz aus! Formulieren Sie die Anforderung ansonsten so, dass sie aus genau einem Hauptsatz besteht!

Definitionen (Regel 23, 24, 25)

Ziel: Die Bedeutung der Begriffe in der Anforderung ist klar.

- Untersuchen Sie jedes Substantiv, ob dieses ein Synonym zu einem Glossareintrag darstellt! Wenn ja, ersetzen Sie es mit dem Glossareintrag und ergänzen Sie diesen um das ersetzte Wort!

- ◇ Untersuchen Sie jedes Substantiv, ob dieses erklärungsbedürftig ist! Wenn dies der Fall ist und kein Glossareintrag dafür existiert, erzeugen Sie einen!
- ◇ Führen Sie die beiden Schritte für jedes Verb durch!
- ◇ Führen Sie die beiden Schritte für jedes Adjektiv durch!

Informationen (Regel 3, 4, 5, 10, 12, 13, 14, 15)

Ziel: Die Anforderung enthält ausreichende detaillierte Informationen, um die geforderte Funktionalität umsetzen zu können.

- ◇ Untersuchen Sie die Anforderungen auf die nachfolgenden Aspekte. Notieren Sie sich dabei Rückfragen zur Klärung mit dem Anforderungssteller!
- ◇ Untersuchen Sie ob eine Möglichkeit in der Anforderung ausgedrückt wird und überprüfen Sie, ob die Schritte zur Umsetzung dieser Möglichkeit klar sind!
- ◇ Untersuchen Sie, ob technische Details zur Umsetzung einer Funktionalität gegeben werden und entfernen Sie diese, falls sie nicht ausdrücklich gefordert werden!

Verben

- ◇ Bestimmen Sie die Verben in der Anforderung!
- ◇ Untersuchen Sie, ob ein Satz der mehrere Akteure enthält mit diesen Verben vorstellbar wäre!

Adjektive

- ◇ Bestimmen Sie alle Adjektive in der Anforderung!
- ◇ Wenn diese einen Vergleich/eine Steigerung ausdrücken, untersuchen Sie, ob ein Bezugspunkt gegeben ist!
- ◇ Stellen Sie sich die Frage ob, wie und mit welcher Genauigkeit die Eigenschaft messbar sind!
- ◇ Untersuchen sie, ob klar ist, was das System zu erbringen hat, um die Eigenschaft zu erfüllen!

Substantive

- ◇ Bestimmen Sie alle Substantive in der Anforderung!
- ◇ Setzen Sie jedes in den Singular, sofern es sich nicht auf eine Gruppe in Ihrer Gesamtheit bezieht.

- ◇ Untersuchen Sie, ob es sich um eine Person, Personengruppe, ein Objekt oder eine Objektgruppe handelt! Ergänzen Sie die Anforderung jeweils so, dass klar wird „Wer", „Was" bzw. „Welcher Teil der Gruppe" mit dem Substantiv umschrieben wird.
- ◇ Überprüfen Sie, ob eine Mengenangabe („genau ein", „jeder", „alle") vor dem Substantiv sinnvoll ist. Falls ja, ermitteln Sie die Menge. Falls nein, verwenden Sie den bestimmten Artikel vor dem Substantiv!

Sonderfälle (Regel 6, 9, 10, 11)

Ziel: Es werden alle mit der Anforderung verbundenen Sonderfälle berücksichtigt.

- ◇ Untersuchen Sie, ob aus der Anforderungen explizit (Universalquantoren) oder implizit ersichtlich wird, für wen und wann die Funktionalität gelten soll!
- ◇ Ergänzen Sie diese Informationen durch definierte quantitative Angaben!
- ◇ Überlegen Sie ob ein Ausnahme- oder Fehlerfall zur Anforderung spezifiziert werden muss!
- ◇ Überprüfen Sie, ob Bedingungen innerhalb der Anforderung ausgedrückt werden und falls ja, ob diese alle möglichen Fälle abdecken!
- ◇ Überprüfen Sie, ob die geforderte Funktionalität selbst nur unter bestimmten Bedingungen gilt und ergänzen Sie die Sonderfälle!

Implizite Anforderungen (Regel 7, 8)

Ziel: Es werden alle Anforderungen erkannt, die zur Erfüllung dieser Anforderung nötig sind.

- ◇ Stellen Sie sich die Frage, welche Aussagen aus der Anforderung hervorgehen, die erfüllt werden müssen, damit die Anforderung umgesetzt werden kann!
- ◇ Überprüfen Sie, ob das Verhältnis zwischen mehreren Objekten etwas Wesentliches darstellt. Falls ja, formulieren Sie dafür eine eigene Anforderung!

Umgeformte Anforderung

Text der umgeformten Anforderung

Validation (Regel 16, 23, 24, 25)

- ◇ Legen Sie die Originalanforderung und die umgeformte Anforderung dem Anforderungssteller vor und lassen Sie sich die Korrektheit der Umformulierung bestätigen!
- ◇ Definieren Sie zusammen mit dem Anforderungssteller die Glossareinträge, die aus der Anforderung entstanden sind! Verwenden Sie dazu folgende Schemata:
  - ◇ Substantiv: „Ein“ + Subjekt (=zu definierender Begriff) + Verb + Objekte + Ergänzungen (Phrasen, Nebensätze)
  - ◇ Verb: Verb im Infinitiv + "ist der Prozess" + Ergänzungen
  - ◇ Adjektiv: Zu definierendes Eigenschaftswort + Hilfssubstantiv + "ist" + Hilfssubstantiv + Ergänzungen
- ◇ Klären Sie die entstanden Rückfragen mit dem Anforderungssteller!

Neuformulierung

- ◇ Entscheiden Sie aus den Antworten des Anforderungsstellers, ob neue Anforderungen zu erstellen sind. Formulieren Sie die neue(n) Anforderung(en)!

Ergebnisanforderung

Ergebnisanforderung

Qualitätssicherung

- ◇ Wenden Sie das Regelwerk auf die neue(n) Anforderung(en) an!

9 Zusammenfassung und Fazit

Zum Ende der Arbeit sollen die wesentlichen Punkte der Bewertung nochmals zusammengefasst werden. Da bereits zum Abschluss der beiden Bewertungskapitel 6 und 7 Zusammenfassungen vorgenommen wurden, werden hier nur die Kernpunkte der Bewertung aufgegriffen.

Ein wesentliches Ergebnis der Bewertung besteht darin, dass mit dem Regelwerk auf klare Art und Weise die Qualität von Anforderungen sichergestellt werden kann. Die 25 Regeln sind dafür bis auf wenige Ausnahmen gut geeignet.

Zu kritisieren sind die ungenaue, schwer verständliche Darstellung des Regelwerks und das Fehlen einer Vorgabe, wie das Regelwerk schrittweise auf Anforderungen angewendet werden soll. Dies sind jedoch zwei Kritikpunkte, die durch eine Überarbeitung des Regelwerks entfernt werden können (und durch den Leitfaden für die Anwendung bereits abgeschwächt wurden). Ebenso ist zu bemängeln, dass das Regelwerk schlecht in die weiteren Aufgaben im Rahmen der Systemanalyse integriert ist. Auch hierfür sind jedoch Lösungen denkbar.

Als Hauptkritikpunkt ist der große Zeitaufwand zur Einarbeitung in das Regelwerk und insbesondere zu dessen Anwendung zu sehen. Hierbei ist davon auszugehen, dass dieser dem Regelwerk inhärent ist und er auch durch die Optimierung des Regelwerks und Einschränkungen in der Anwendung nicht wesentlich vermindert werden kann.

Diese Problematik geht soweit, dass die Praxistauglichkeit des Regelwerks in seiner derzeitigen Form stark angezweifelt werden muss. Gestützt wird diese Einschätzung dadurch, dass weder umfangreiche Schulungen noch unterstützende Werkzeuge für das Regelwerk angeboten werden. Auch scheint der Verbreitungsgrad in der Wirtschaft vergleichsweise gering zu sein.

Der hohe Zeitaufwand kann deutlich reduziert werden, wenn Anforderungen mit hoher Ausgangsqualität vorliegen und wenn eine Reihe weiterer - in diesem Bericht diskutierter - Rahmenbedingungen erfüllt sind, die für einen effizienten Einsatz des Regelwerks nötig sind. Alternativ kommt in Betracht nur ausgewählte Teile des Regelwerks anzuwenden und sich auf die Grundaussagen zu beschränken. Der umfassende Charakter des Regelwerks geht in diesem Fall jedoch zwangsläufig verloren.

Trotz dieser Einschränkungen ist es zu empfehlen, das Regelwerk in die Lehre an einer Berufsakademie im Rahmen einer Veranstaltung aufzunehmen, in der auch andere Methoden des Requirements Engineerings behandelt werden. Auch wenn die Studierenden das Regelwerk später nicht in seiner Vollständigkeit anwenden, dürften sie von dem Bewusstsein über Fehlerquellen profitieren und können Anforderungsanalysen bereits durch Beachten der Kernaussagen verbessern.

Anhang

A Das SOPHIST-REgelwerk im Überblick

1: Formulieren Sie jede Anforderung im Aktiv
Die Aktivformulierung hat den Vorteil, dass der Täter, also die ausführende Person oder Einheit, in der Anforderung angegeben werden muss. Dies ist gerade bei Anforderungen entscheidend, da hier wichtig ist, ob die Aktivität vom System, vom Nachbarsystem oder vom Benutzer durchgeführt wird. Auf diese Weise erhält die Anforderung einen höheren Informationsgehalt und das Prozesswort wird näher spezifiziert, da angegeben wird, wer den Prozess durchführt.

2: Drücken Sie Prozesse durch Vollverben aus
Jeder Prozess sollte durch ein Vollverb ausgedrückt werden. Adjektive oder aufwändige Phrasen verschleiern nur den Prozess und lassen die eigentlich durch die Anforderung geforderte Funktionalität in den Hintergrund treten. Vollverben verlagern zudem weitere Satzbestandteile, damit sie näher spezifiziert sind.

3: Decken Sie unvollständig spezifizierte Prozesswörter auf
Bestimmen Sie die Verben in einer Anforderung und stellen Sie fest, ob ein Satz, der mehrere Akteure enthält, mit diesen Verben vorstellbar wäre. Ist dies der Fall, so ist aus dem Originalsatz Information getilgt worden. Ist die fehlende Information wissenswert, sollten Sie gezielt danach fragen.

4: Ermitteln Sie unvollständige Vergleiche und Steigerungen
Bestimmen Sie die Vergleiche und Steigerungen in einer Anforderung. Enthalten sie den Bezugspunkt, auf den sich der Vergleich oder die Steigerung bezieht? Ist die Abweichung überhaupt messbar? Wenn ja, mit welchen Mitteln (Messmethode)? Mit welcher Genauigkeit kann gemessen werden?

5: Klären Sie Mögliches und Unmögliches
Stellen Sie bei Aussagen, die eine Möglichkeit oder auch eine Unmöglichkeit beschreiben (siehe Signalwörter), folgende Frage: Was macht das genannte Verhalten möglich bzw. unmöglich? Signalwörter: „kann", „darf", „es ist möglich", „es ist außerstande"

6: Prüfen Sie Modaloperatoren der Notwendigkeit auf benötigte Ergänzungen
Überlegen Sie sich zu jeder durch einen Modaloperator der Notwendigkeit (siehe Signalwörter) spezifizierten Aussage, ob Sie zusätzlich ein Verhalten für den Ausnahmefall spezifizieren müssen. Signalwörter: „müssen", „sollen", „sollte", „es ist notwendig"

7: Finden Sie implizite Annahmen
Bestimmen Sie das Hauptverb eines Satzes und bilden Sie eine neue Oberflächenstruktur durch Negation dieses Verbs. Danach fragen Sie sich, welche Aussagen wahr sein müssen, damit beide Oberflächenstrukturen einen Sinn ergeben. Alle Aussagen, die Sie dabei entdecken werden, sind unter Umständen nicht formulierte Annahmen, die geklärt und expliziert werden müssen.

8: Überprüfen Sie das Verhältnis der Objekte
Überprüfen Sie, ob das Verhältnis zwischen mehreren Objekten etwas Wesentliches darstellt. Falls dies der Fall ist, formulieren Sie für das Verhältnis eine eigene Anforderung. Falls dies nicht der Fall ist, formulieren Sie für das Verhältnis einen einzigen Begriff (geschlossene Einheit). Der verwendete Begriff ist dann allerdings meist eine Nominalisierung, zu der eine grobe Definition hinterlegt sein muss.

9: Bestimmen Sie Universalquantoren
Bestimmen Sie die Universalquantoren einer Anforderung und fragen Sie jeweils, ob das geforderte Verhalten des Systems für wirklich alle Objekte aus der Menge gelten soll, die durch den Quantor zusammengefasst werden. Vielleicht gibt es Ausnahmen, die Sie zusätzlich spezifizieren müssen. Untersuchen Sie jeden Satz dahingehend, ob eine Menge an Objekten ausdrücklich definiert ist, für die das spezifizierte Verhalten gelten soll. Signalwörter: „nie“, „immer“, „kein“, „jeder“, „alle“, „irgendeiner“, „nichts“

10: Verwenden Sie nur definierte quantitative Angaben
Verwenden Sie als quantitative Angaben zum Beispiel nur "alle", "jeder/jedem", "entweder", "immer", "oder", und "kein" im Deutschen.

11: Ermitteln Sie unvollständig spezifizierte Bedingungen
Bestimmen Sie die Bedingung(en) in der Anforderung und überprüfen Sie, ob sowohl für den Fall, dass die Bedingung eintritt, ein Verhalten spezifiziert ist (dann-Zweig), als auch dafür, dass sie nicht eintritt (sonst-Zweig). Stellen Sie sich zudem die zusätzlichen Fragen: "Sind alle möglichen Entscheidungskriterien (Bedingungen) aufgezählt?" und "Sind alle möglichen Varianten geschildert?" Signalwörter: „wenn“, „dann“, „falls“, „im Falle von“, „abhängig von“

12: Hinterfragen Sie Substantive ohne Bezugsindex
Fragen Sie für jedes Substantiv der Anforderung, ob es eigentlich eine spezifische Person, eine spezifische Personengruppe, einen spezifischen Gegenstand oder eine Gruppe von Gegenständen der Welt bezeichnen sollte. Sie können bei Substantiven, die eine nicht genau einzugrenzende Menge von Objekten beschreiben, die folgenden Fragen stellen: "Wer ... genau?", "Was ... genau?", "Welcher Teil der genannten Menge?" Erweitern Sie solche Substantive dann durch eine genau festlegende Ergänzung.

13: Verwenden Sie Substantive stets in der Einzahl
Die in der Anforderung vorkommenden Substantive sollten zudem in der Einzahl verwendet werden, außer der geforderte Sachverhalt bezieht sich ausschließlich auf eine Gruppe in ihrer Gesamtheit.

14: Machen Sie konkrete Angaben über die Menge an Objekten
Falls es sinnvoll ist, sollten Sie den Artikel je nach Bedeutung durch "jeder", "alle" oder "genau ein" ersetzen. Angaben über die Menge der Objekte, die Substantive umschreiben, werden nicht nur durch Mehr- oder Einzahl ausgedrückt, sondern durch spezifische Angaben, die den Artikel (der, die, das, ein, eine, ein, ...) vor dem Substantiv ersetzen, noch näher spezifiziert.

15: Verwenden Sie unbestimmte Artikel nur in Definitionen
Verwenden Sie ein, ein (unbestimmter Artikel) nur vor einem Substantiv, das damit gerade definiert wird.

16: Verwenden Sie den bestimmten Artikel vor einem Substantiv, das schon definiert ist
Ist keine Mengenangabe vor einem Substantiv notwendig oder sinnvoll, sollte in einer Anforderung immer der bestimmte Artikel (der, die, das oder eine deklinierte Form wie dem, den, das) verwendet werden

17: Hinterfragen Sie Nominalisierungen
Überprüfen Sie jedes Substantiv im Satz und fragen Sie sich, ob nicht ein Verb möglich gewesen wäre, das einen Vorgang beschreibt und das ähnlich klingt/aussieht und dem Substantiv in der Bedeutung ähnelt. Dazu zwei Tests: 1. Passt das Substantiv sinnvoll in die Phrase "ein(e) andauernd(r) ..." (im Sinne von kontinuierlich), so handelt es sich um einen Vorgang, der normalisiert wurde. 2. Beschreibt ein Substantiv etwas, dass man nicht anfassen kann? Auch hier handelt es sich wahrscheinlich um einen nominalisierten Prozess. Kann eine der beiden Fragen mit "Ja" beantwortet werden, so müssen Sie das Substantiv dahingehend hinterfragen, ob nicht mit dem unterschlagenen Vorgang auch Informationen verloren gegangen sind.

18: Ersetzen Sie die Funktionsverbgefüge durch einfache, direkte Vollverben
Funktionsverbgefüge führen zu einem mit Substantiven überladenen, schwer verständlichen Stil und bewirken oft, dass ein Vorgang in passivischer Sichtweise beschrieben wird. Viele Sätze sind unnötig indirekt ausgedrückt, sodass die eigentliche Tätigkeit, die normalerweise durch ein Vollverb beschrieben ist, in den Hintergrund gerät.

19: Vermeiden Sie Redundanz
Vermeiden Sie es, etwas doppelt auszudrücken. Entfernen Sie Teile des Satzes, die Sie ohne Bedeutungsverlust straffen können.

20: Vermeiden Sie Floskeln
Streichen Sie floskelhafte Wörter und Wendungen oder drücken Sie diese kürzer aus.

21: Überprüfen Sie Nebensätze
Überprüfen Sie Nebensätze, die eine Begründung, Absicht oder Folge enthalten. Falls darin keine wichtigen Informationen verborgen sind, können Sie diese als Kommentar herauslösen, ansonsten müssen Sie daraus eigenständige Anforderungen formulieren.

22: Stellen Sie zeitliche Abhängigkeiten klar
Nebensätze, die in einem zeitlichen Verhältnis zum Hauptsatz stehen, müssen Sie in einem Wenn/Falls-Satz mit eventuellen Verneinungen umformulieren. Schlüsselwörter zur Identifizierung temporaler Nebensätze (Nebensätze, die in einem zeitlichen Verhältnis stehen) sind zum Beispiel: "bevor", "während", "nachdem", "solange", "bis", "unterdessen", "solange", "bis", "unterdessen".

23: Definieren Sie Substantive nach folgendem Schema
Subjekt (=zu definierender Begriff) + Verb + Objekte + Ergänzungen (Phrasen, Nebensätze)

24: Definieren Sie Eigenschaftswörter nach folgendem Schema
Zu definierendes Eigenschaftswort + Hilfssubstantiv + "ist" + Hilfssubstantiv + Ergänzungen

25: Definieren Sie Verben nach folgendem Schema
Verb im Infinitiv + "ist der Prozess" + Ergänzungen

Literaturverzeichnis

(Bandler/Grinder75)

Richard Bandler, John Grinder (1975): The Structure of Magic II, Palo Alto/CA (Science and Behaviour Books)

(Bröhl/Dröschel99)

Adolf-Peter Bröhl, Wolfgang Dröschel (1999): Das V-Modell. 2. Auflage. München, Wien (Oldenbourg)

(Ebert05)

Christof Ebert (2003): Systematisches Requirements Management - Anforderungen ermitteln, spezifizieren, analysieren und verfolgen. Heidelberg (dpunkt Verlag)

(Havemeister03)

Thomas Havemeister (2003): Konzeption eines Prozessmodells zum Requirements Engineering für individuelle Softwarelösungen, http://www.theoinf.tu-ilmenau.de/~streitdf/TheHome/DA_SA/Data/DA_Havemeister_Thomas.pdf

(IEEE98)

IEEE Standard 830 (1998): IEEE Standard 830-1998. IEEE Guide for Information Technology – System Definition, New York, NY, USA (IEEE)

(IFI07)

IFI Requirements Engineering Group (2007): http://www.ifi.unizh.ch/rerg/teaching/courses/requirements_engineering_ii

(Kof05)

Leonid Kof (2005): Text Analysis for Requirements Engineering, http://www4.in.tum.de/~kof/publications/Leonid_Kof_Text_Analysis_Diss.pdf

(Richter98)

Martin Richter (1998): Konzeption und Implementierung von Verfahren zur (halb-)automatischen Prüfung natürlichsprachlicher Spezifikationen, http://elib.uni-stuttgart.de/opus/volltexte/1999/355/pdf/355_1.pdf

(Robertson/Robertson06)

Suzanne Robertson, James Robertson (2006): Requirements Management: a Cinderella Story, http://www.volere.co.uk/-cinderella.htm (abgerufen am 21.11.2006)

(Rupp02)

Chris Rupp (2002): Requirements-Engineering und –Management. Professionelle, iterative Anforderungsanalyse für die Praxis, 2. Auflage, München Wien (Carl Hanser Verlag)

(Rupp07)

Chris Rupp (2007): Requirements-Engineering und –Management. Professionelle, iterative Anforderungsanalyse für die Praxis, 4. Auflage, München Wien (Carl Hanser Verlag)

(Ryser03)

Johannes Ryser (2003): Szenarienbasiertes Validieren und Testen von Softwaresystemenm, http://www.ifi.unizh.ch/archive/diss/Jahr_2003/thesis_ryser.pdf

(SOPHIST01)

SOPHIST GmbH (2001): Die psychotherapeutischen Grundlagen des SOPHIST-REgelwerks für die natürlichsprachliche Metode, http://www.sophist.de ! Downloads ! öffentlicher Downloadbereich (Registrierung nötig), abgerufen am 20.11.06

(SOPHIST07)

SOPHIST-Group (2007), Requirements Engineering in der Praxis: Anforderungen mit Prosa und Modellen clever erheben und dokumentie-

ren, http://www.sophist.de/Trainings.nsf/Lookup/3C48B564395D2B81C1256EBE002E3AE7?Open, abgerufen am 9.2.2007

(TUM07)

TU München (2007): Hauptseminar Sommersemester 2004 Requirements-Engineering, http://www4.in.tum.de/lehre/seminare/hs/SS04/requirements/